Servidor en Linux Debian

Manual original, creado por **David Sanchez Cantero**
Mas conocido en la red por **Forat**
ForatDot.Info

¿ Te animas a montarte tu propio servidor Web ?

Si te gustaría tener un servidor Web y un **Blog** como este con tus propios contenidos y sin censura alguna, ahora tienes una oportunidad única. Se trata de 14 capítulos que explican al detalle cosas como por ejemplo la configuración del Hardware, instalación del sistema operativo **Linux Debian**, **Servidor Web Apache 2** compatible con **PHP**, **Proftpd**, **PhpSysInfo**, **AwStats**, bases de datos **MySQL**, **PhpMyAdmin**, **apuntes**, **consejos** y todo lo que se me a ocurrido por el camino.

A continuación podéis ver la lista de capítulos …

Introducción

- Vol 1 (**Hardware)**
- Vol 2 (**Instalación del sistema operativo Linux Debian)**
- Vol 3 (**Configuración de Red)**
- Vol 4 (**Acceso remoto con SSH (Secure Shell))**
- Vol 5 (**Apache Web Server)**
- Vol 6 (**Soporte para PHP)**
- Vol 7 (**Bases de datos MySQL y phpMyAdmin)**
- Vol 8 (**Información del sistema con PhpSysInfo)**
- Vol 9 (**File Transfer Protocol un FTP con ProFtpd)**
- Vol 10 (**Estadísticas Web con AwStats)**
- Vol 11 (**Redireccionamiento DNS de No-IP)**
- Vol 12 (**Abriendo puertos en nuestro Router)**
- Vol 13 (**Últimos ajustes)**
- Vol 14 (**Instalación y puesta en marcha de un Blog en Wordpress)**

Los que me siguen saben que siempre que monto un servidor también monto un Blog nuevo para el con un nuevo diseño y contenido. Esta vez no va a ser menos y podemos verlo en la dirección ...

http://hsnet.sytes.net

Espero que este y los proyectos que nos esperan os sean de gran ayuda amigos.

Introducción

Así es amigos, si os gustaría tener un servidor web casero en vuestra propia casa próximamente podréis leer en este **Blog** en forma de varios capítulos un manual detallado sobre como montar un servidor web casero de la manera mas fácil y sencilla posible. Como de costumbre pienso explayarme ilustrando los artículos con fotografías y explicaciones al detalle …

Desde hoy todos los que vais a montaros esté servidor web vais a poder gozar de una herramienta con la que no cuenta todo el mundo. Vais a poder acceder desde Internet a vuestro servidor que tenéis en casa, y esto *es una pasada* (*pensareis muchos*) pero todo tiene su lado oscuro y es que tenéis que pensar que al leer webs o blogs, o subir archivos o bajarlos por el FTP tendréis que tener ancho de banda para poder hacerlo ya que si la tenéis ocupada por

programas P2P el tiempo de respuesta puede ser eterno o nulo.

De este nuevo servidor que yo mismo iré montando a tiempo real podremos sacar un partido limitado y digo esto porque no todo va a ser un camino de rosas …. Siempre tenemos que contar que no necesitamos un ordenador super potente para montar un servidor pero cuanto mas potente sea mas deprisa irán las cosas como todo en la vida. **(Aquí podéis ver las señales vitales sobre todo el hardware de el servidor que tengo en mi casa a tiempo real**)

También tenemos que contar como punto negativo que cuando conectemos este servidor a nuestra ADSL contaremos con el triste embudo de nuestro ancho de banda.

Explicados estos puntos me gustaría que os quedara claro que cuanto mas potente sea el PC que usemos mejor (*aunque no sea necesario*) en el primer articulo explicare que piezas son las que necesitamos cuales son mas potentes. Por otro lado también os aclaro que cuanto mas grande sea vuestro ancho de banda mejor y si tiene mucho ancho de banda de subida mejor que mejor. Pensad que para descargar cosas de programas P2P o simplemente navegar necesitamos cuanta mas tasa de descarga mejor pero para que los usuarios de Internet puedan usar los recursos de nuestro servidor necesitamos cuanta mas tasa de subida mejor.

Este mi **Blog** nació en un servidor que tenia **montado en FreeBSD** y funcionaba a las mil maravillas, luego monté otros servidores probando varias *distribuciones de Linux* y finalmente me quede con **Gentoo** pero tuve que dejar de hospedarlo aquí por la limitación del ancho de banda tan pequeño que me ofrece mi **ADSL**. Cuando todo empezó este **Blog** apenas tenia visitas y las demás *webs* que hospedaba tampoco pero con el paso del tiempo empezaron a llegar muchos mas usuarios y habían noches que no podía ni navegar ya que todo el ancho de banda de mi **ADSL** estaba ocupado por los usuarios que leían mis contenidos **Online**. He cambiado tres veces de hospedaje porque cada día necesito mas recursos, tanto de sistema como de tasa de transferencia y ancho, así que no penséis en crearos este servidor para hospedar un Blog que tendrá muchísimas visitas porque os sentiréis frustrados ya que tener mucho ancho de banda solo están al alcance de grandes empresas y no para mi, pero si que podéis usarlo para empezar a crear uno o varios Blogs, ademas de poder diseñar themes o experimentar con la programación web, subir y bajar archivos a tu

propio ftp tanto desde modo local o desde Internet o conectaros a el por ssh para poder administrarlo remotamente o usarlo de shell, y un sin fin de utilidades mas que os iré explicando en varios capítulos de este manual tan esperado por algunos lectores.

Como siempre voy a intentar que todo sea lo mas comprensible posible para que sin tener mucha idea de esto podáis montaros esta maquina en vuestra casa y paséis un rato divertido aprendiendo con **Linux Debian** y a crear un servidor al que podréis sacarle tanto provecho como imaginación le echéis al asunto.

Hardware

Para empezar este gran manual que recolecta varios manuales en uno voy a montarme un ordenador y os voy a explicar que piezas voy a utilizar y que es lo que necesitamos para montarnos un servidor Web con **Linux Debian** y unos cuantos programas mas.

Antes de seleccionar las piezas que vamos a utilizar debemos pensar en que función es la que va a realizar esta maquina para incluir algunas piezas mas potentes que otras y otras ni siquiera las instalaremos porque no las usaremos.

El uso que le vamos a dar a esta maquina es exclusivamente la de servidor así que voy a prescindir de usar entorno gráfico para ahorrar recursos en nuestra maquina. Esto no quiere decir que no necesitemos una tarjeta gráfica y un monitor si no que no necesitamos una tarjeta gráfica muy potente y el monitor pues da lo mismo ya que operaremos con ella en modo Terminal.

También vamos a prescindir del tema del sonido, por lo que no voy a instalarle ninguna tarjeta de sonido. (*Esto es un servidor no un radiocasette*)

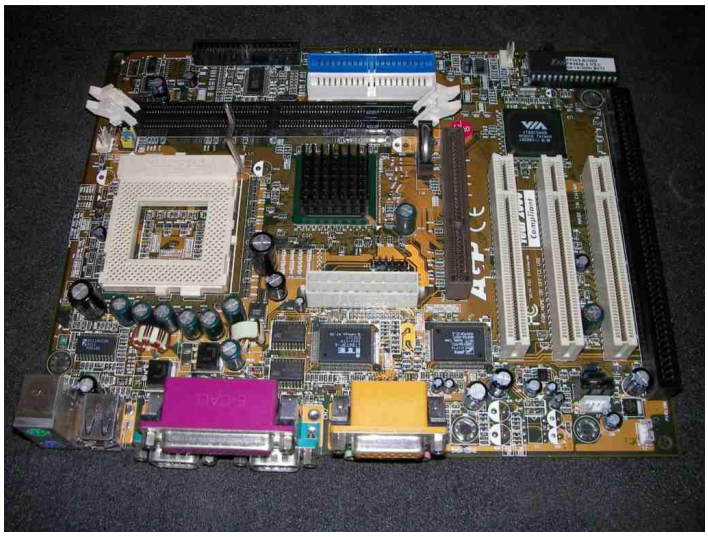
En un servidor Web el **Hardware** mas potente que tiene que haber en la maquina es el microprocesador y la memoria Ram sin duda alguna. De ellos dos depende el funcionamiento de todo una vez empecemos a sobrecargar el sistema con programas y usuarios conectados a el, así que cuanto mas tengamos de esto mejor. En mi caso voy a montármelo en un Pentium III a 800 Mhz con 256 Mb de Ram a 100 Mhz. Con este **Hardware** es suficiente para un servidor casero, funciona a la perfección para uso personal pero no para profesional.

La parte del almacenamiento también es importante pero no tanto como el microprocesador y la memoria. Yo voy a montármelo con un disco duro de 20 Gb y una grabadora de CDs. No nos podemos olvidar de tema de las comunicaciones ya que es importante que incluyamos una tarjeta de red mínimo 10/100 ya que los routers operan a 10/100 habitualmente.

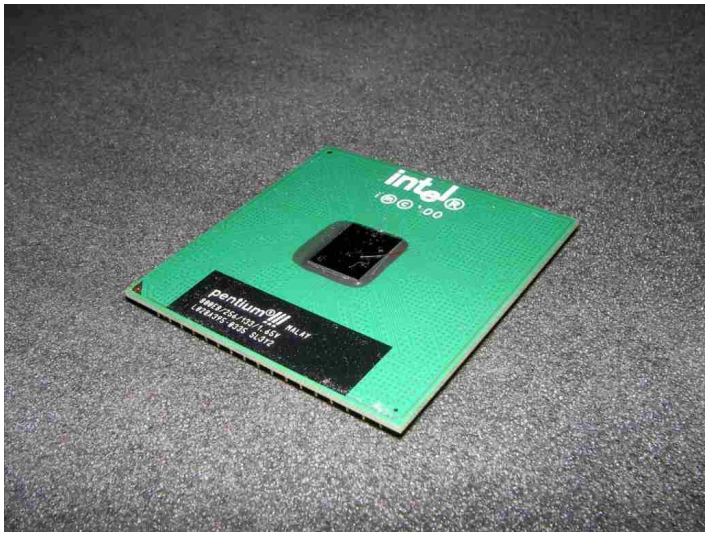
¿ **Que componentes necesitamos** ?

Chasis …

Placa base …



Microprocesador …



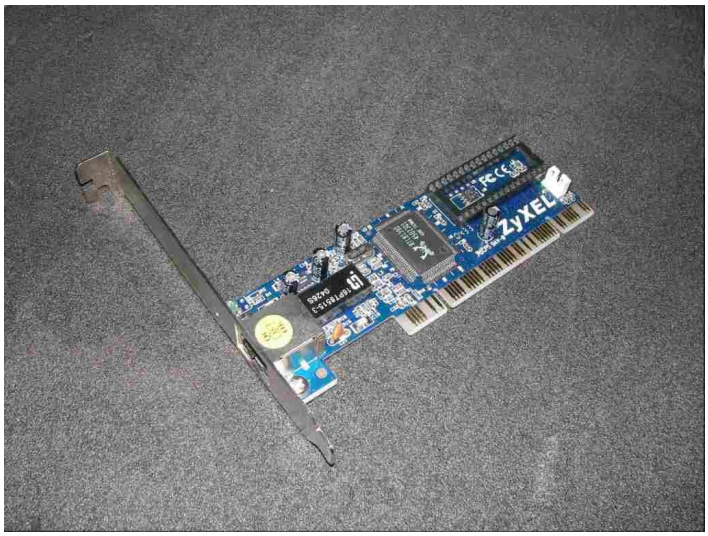
Disipador de calor para el microprocesador …

Modulos de memoria ram …

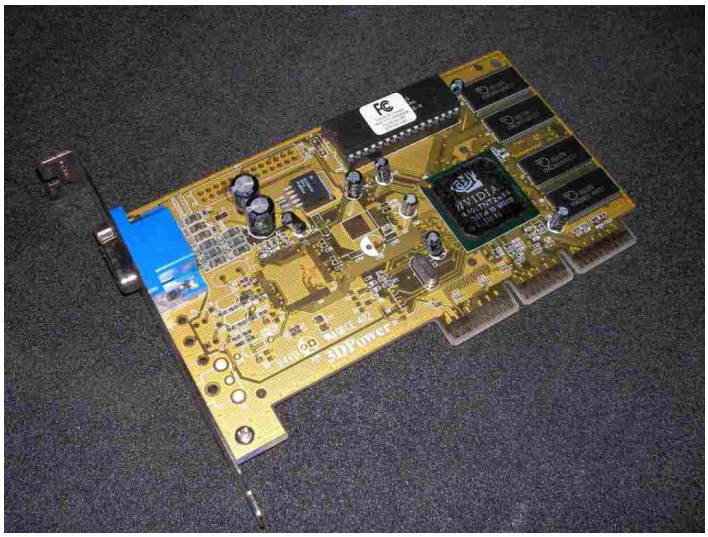
Disco duro …

Unidad grabadora de CD's …

Tarjeta de red …



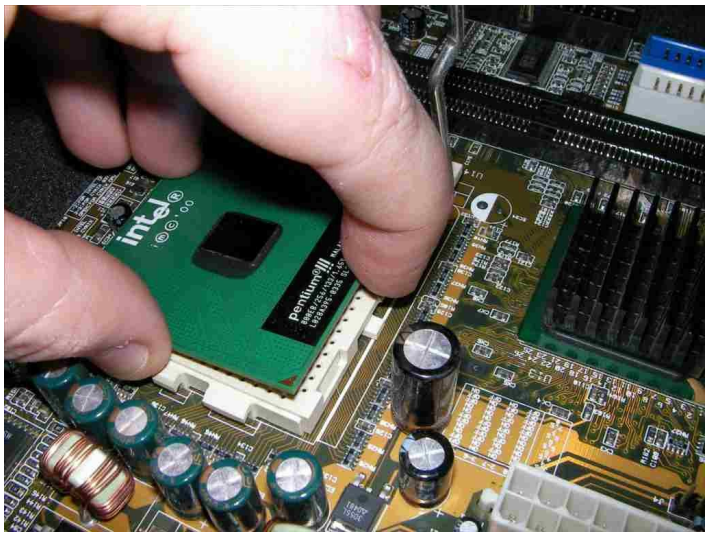
Tarjeta gráfica …



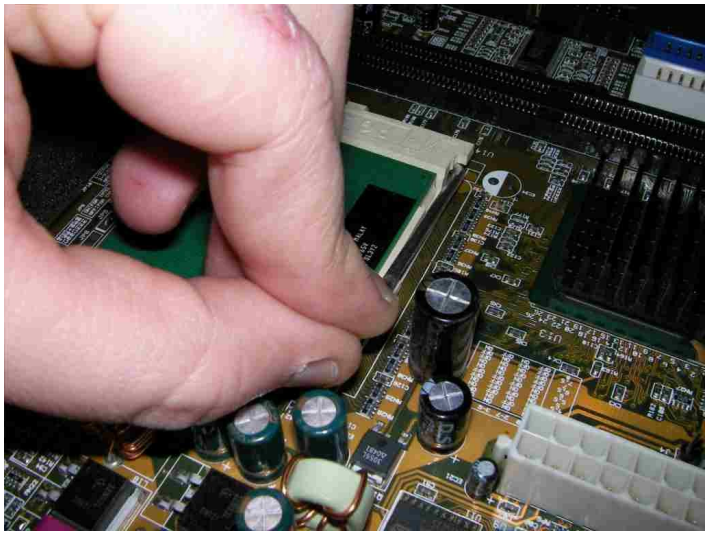
Fuente de alimentación …

Antes de comenzar a encajar todas las piezas dentro de el chasis os recomiendo que instaléis el microprocesador, disipador y la memoria Ram ya que una vez esta todo instalado dentro de el chasis es mas difícil hacer que las piezas estén correctamente fijadas a sus encajes.

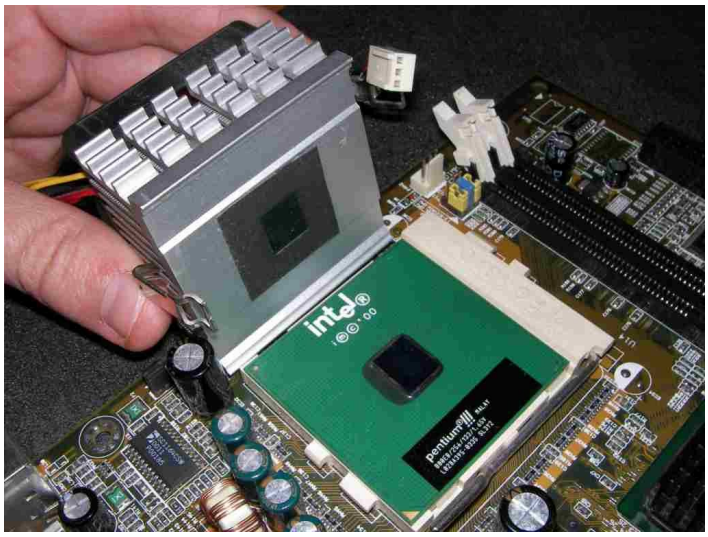
Para comenzar insertamos el microprocesador en su ranura siempre fijándonos en sus laterales que queden correctamente encajados, tienen una mueca para identificar su posición correcta …



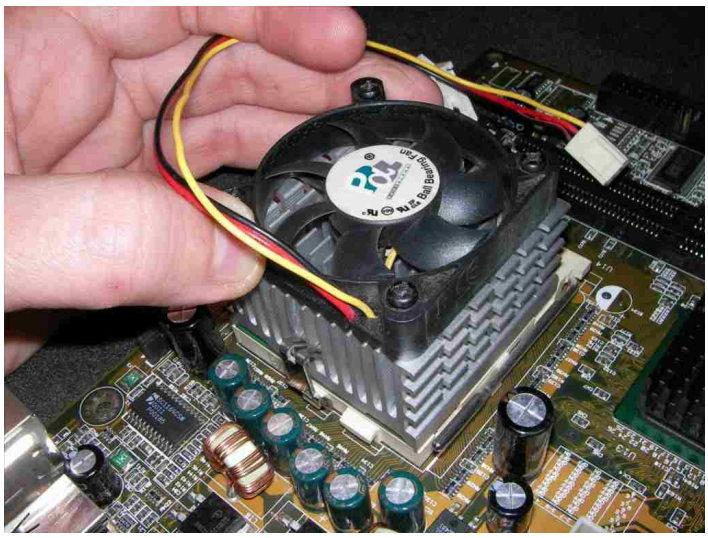
Una vez esté encajado cerramos la maneta para que las patillas de microprocesador hagan contacto con la placa base …



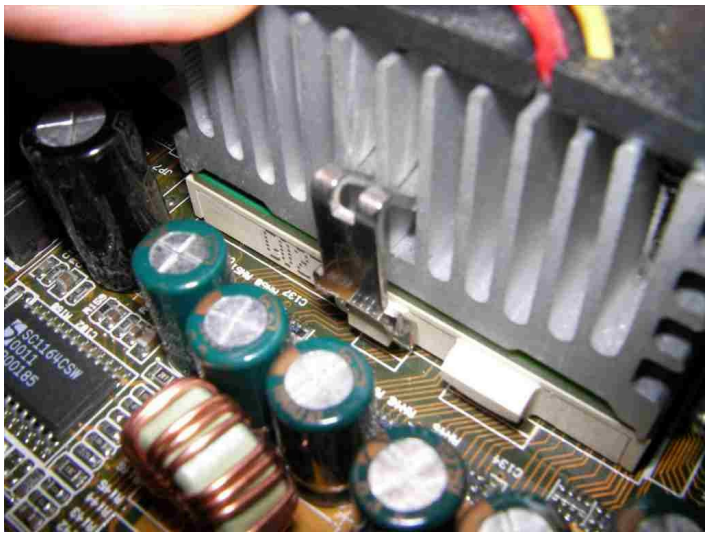
Ahora vamos a instalar el disipador encima del microprocesador, en mi caso no necesita silicona térmica porque viene de serie, si vuestro disipador no incorpora silicona térmica tendréis que ponérsela.



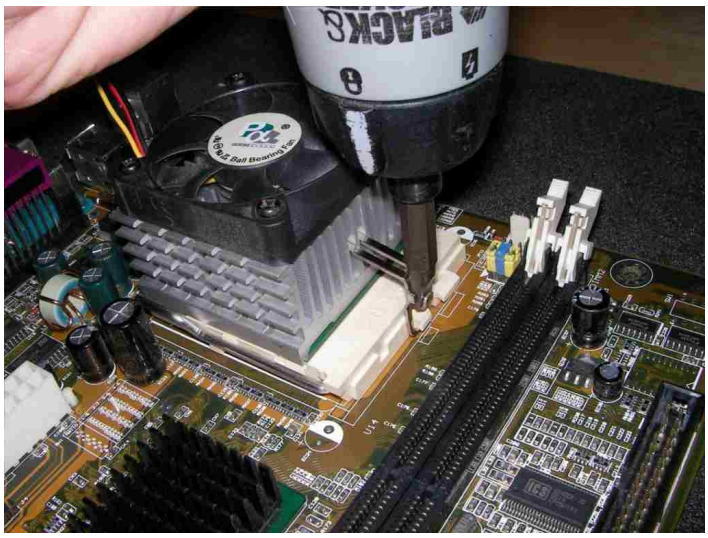
Asentamos el disipador sobre el microprocesador y procedemos a anclarlo a la placa base …



y encajamos primero el soporte no móvil del disipador al anclaje …



Una vez anclado el fijo vamos a por el móvil ayudándonos a encajarlo con un destornillador plano …

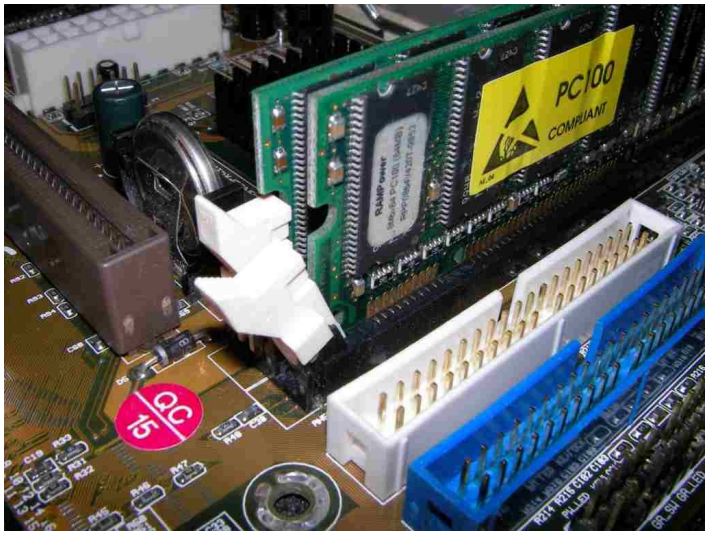


Para mas info *(**Como instalar un microprocesador en una placa base**)

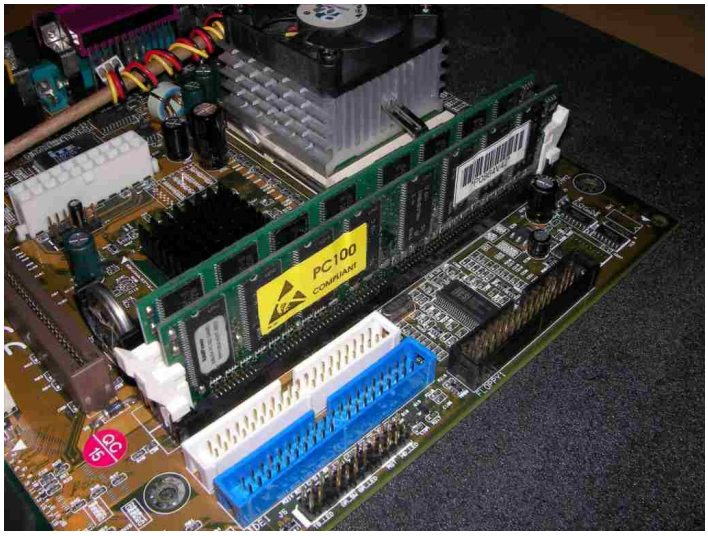
Una vez completados estos sencillos pasos ya tenemos el microprocesador y su disipador correctamente instalados. Con el cable de alimentación del ventilador me he tomado el detalle de enrollarlo en un pincel para que adopte una forma enrollada.

(*Como me gustan estas pijadas*)

Continuamos instalando los módulos de memoria Ram …

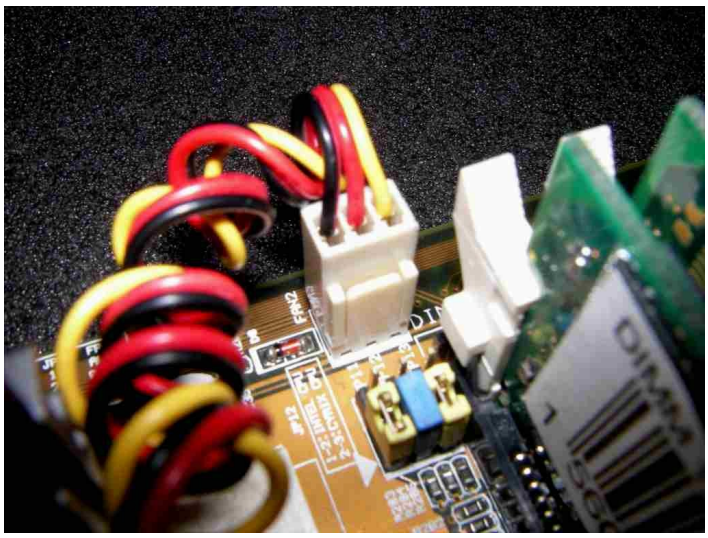


Nos aseguramos de que esa palanquita quede correctamente encajada en el modulo …

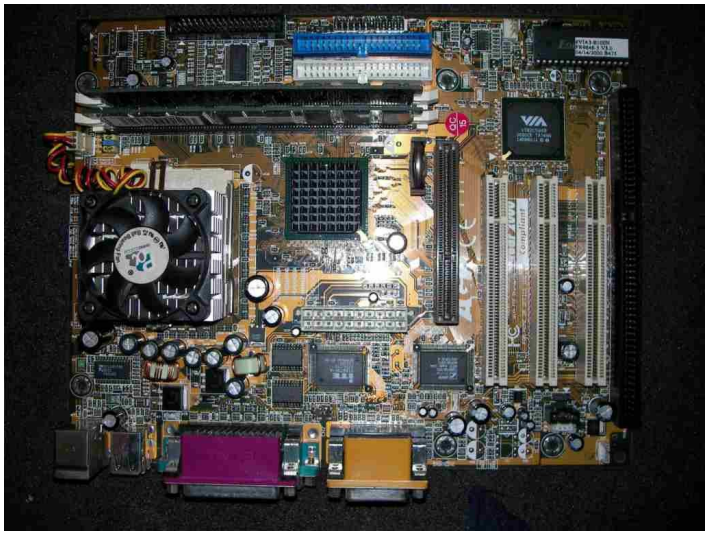


Para mas info *(**Como instalar mas memoria Ram en tu PC**)

Ahora solo nos queda conectar la alimentación a la placa base …



y ya la tenemos lista para instalarla en el chasis …



Si queréis un manual detallado sobre el montaje aquí os dejo un articulo que hice hace tiempo sobre el **montaje de un PC al detalle**.

En el siguiente articulo instalaremos el sistema y lo configuraremos para que tenga conexión a Internet y podamos encontrar nuestro PC desde Internet con un redireccionamiento de **DNS** así que ir preparando vuestras maquinas y afinarlas bien que próximamente vamos a bailar …

Sistema operativo

En este manual podemos ver como instalar **Linux Debian** en unos sencillos pasos, os agradecería a todos los que vais a montaros vuestro servidor casero que sigáis este manual al pie de la letra para asegurarnos de que todos vamos a una. Si tenéis fallos me sera mas fácil detectarlos si habéis hecho lo mismo que yo.

Antes de comenzar con la instalación debéis saber que una vez terminado este manual tendréis el sistema de **Linux Debian** pelado sin gráficos ni ratones, todo en Terminal ya que este PC va a cumplir la función de servidor y no necesita ningún entorno gráfico.

He usado todo el espacio del disco duro ya que no vamos a tener otro sistema operativo en este ordenador. Vamos a dedicar esta maquina para que cumpla sus funciones como servidor y nada mas.

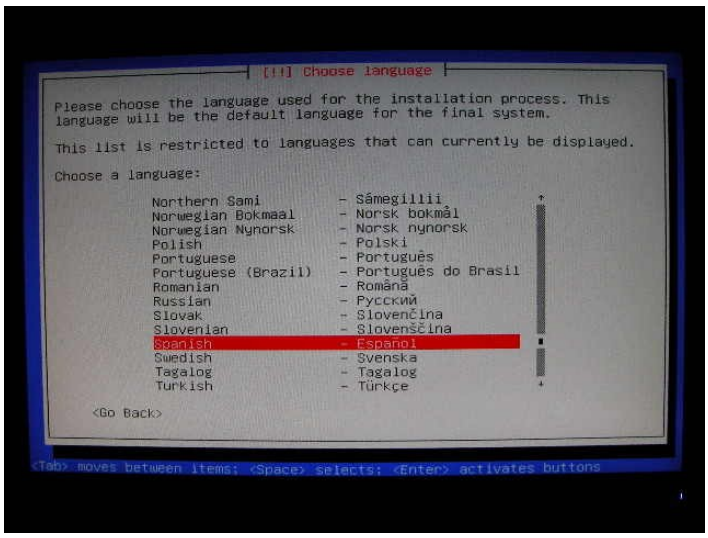
Desde el sistema de particionado del disco duro como la configuración de nuestra red lo aremos de manera automática para no complicaros la vida y no extendernos demasiado. Seguramente en el siguiente articulo hablemos de ello.

Una vez sabidos estos pequeños detalles nos descargamos la ultima versión de **Linux Debian** (la 4.0r5 en estos momentos, **merci captan12** !!) desde el **siguiente enlace**.

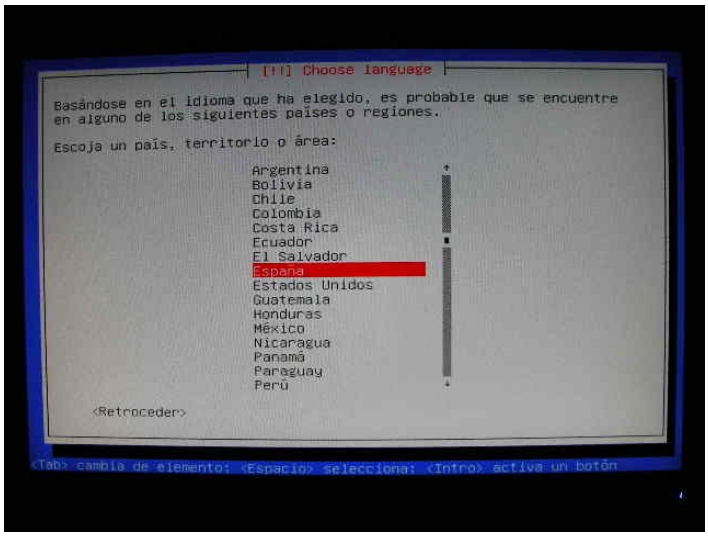
Una vez tengamos la imagen **ISO** la quemamos en un CD y arrancamos nuestro PC con el disco insertado. Comenzamos !!!

En la primera pantalla presionamos (**enter**)

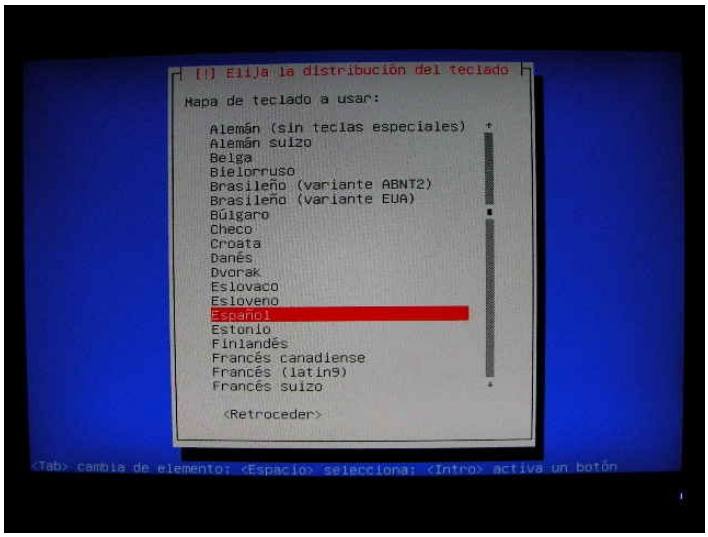
Seleccionamos el lenguaje en el que queremos que se base la instalación …



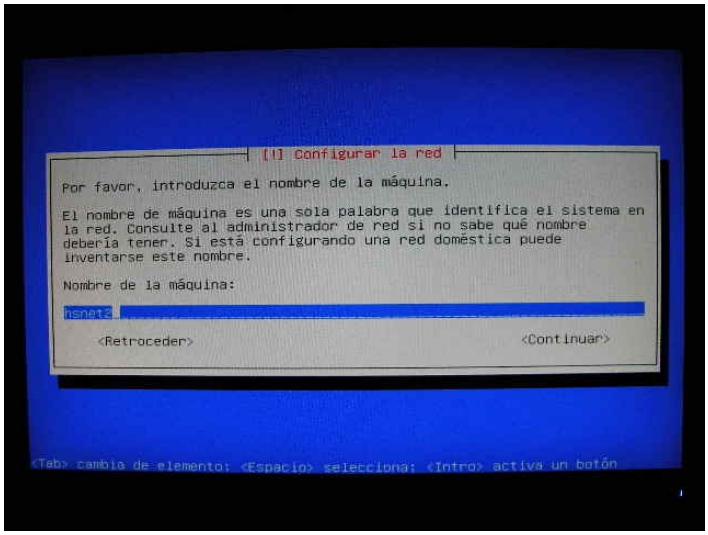
Escoge tu país …



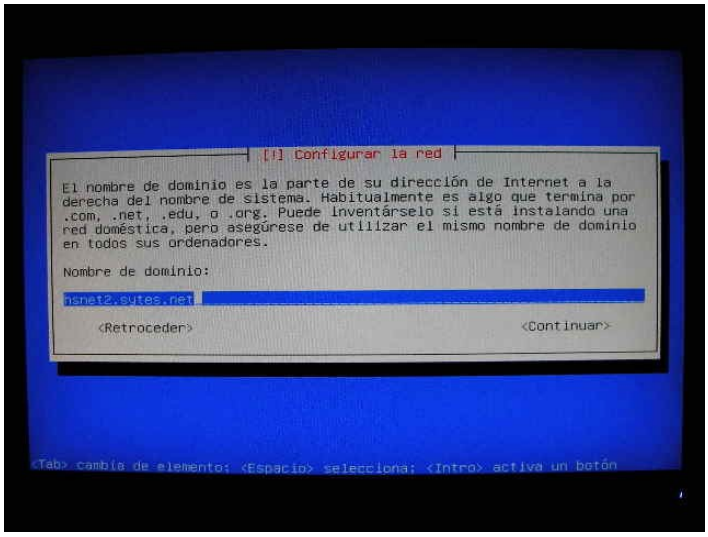
Elije la distribución de tu teclado …



Vamos a darle un nombre a nuestra maquina, en mi caso hsnet2 …

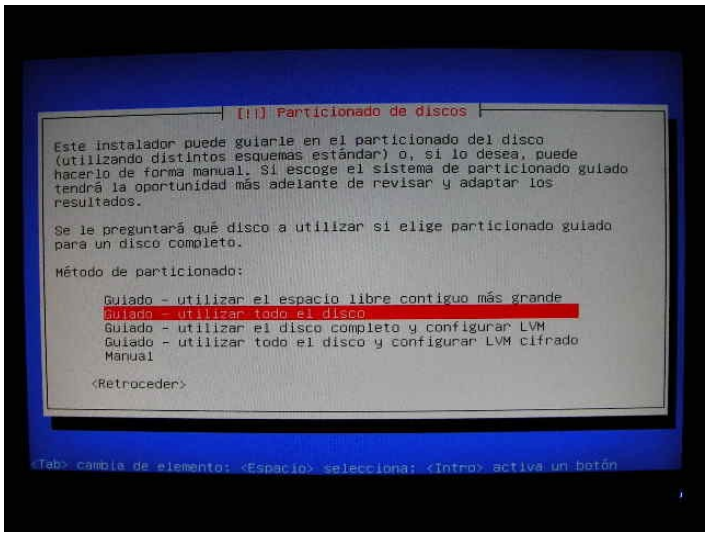


Ahora tenemos que introducir el nombre de dominio. Lo llamaremos .sytes.net con nuestro nombre de maquina delante …

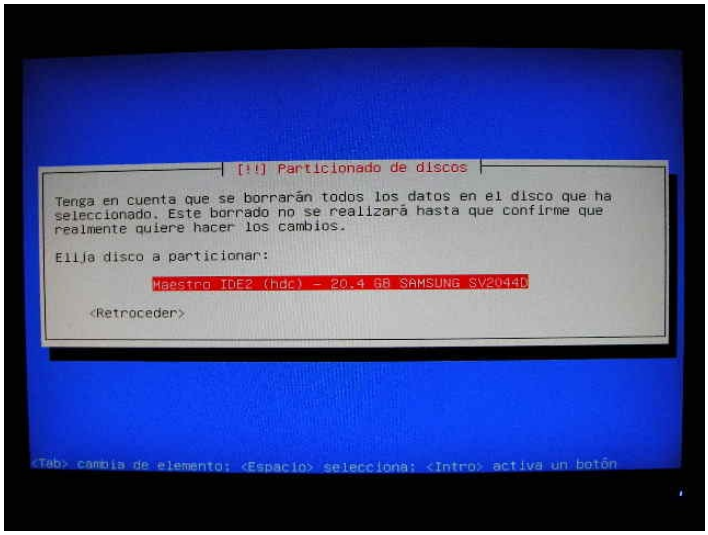


Ahora a llegado el momento del particionado de disco. Como se supone que vamos a usar esta instalación de **Linux Debian** como servidor no compartiremos el disco duro con algún otro sistema así que usaremos todo el espacio posible en disco para la instalación.

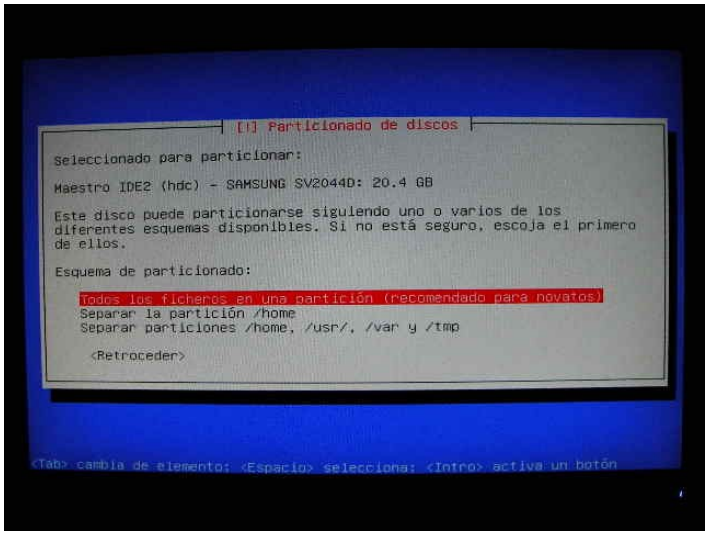
Seleccionamos (**Guiado - utilizar todo el disco**)



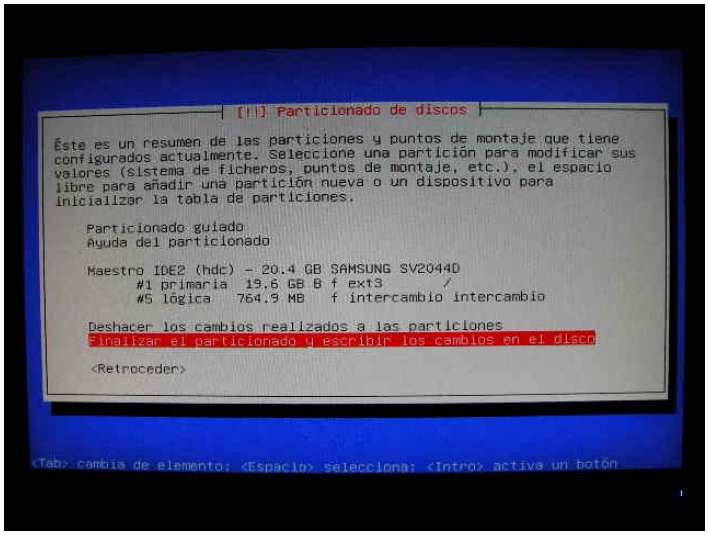
seleccionamos el disco …



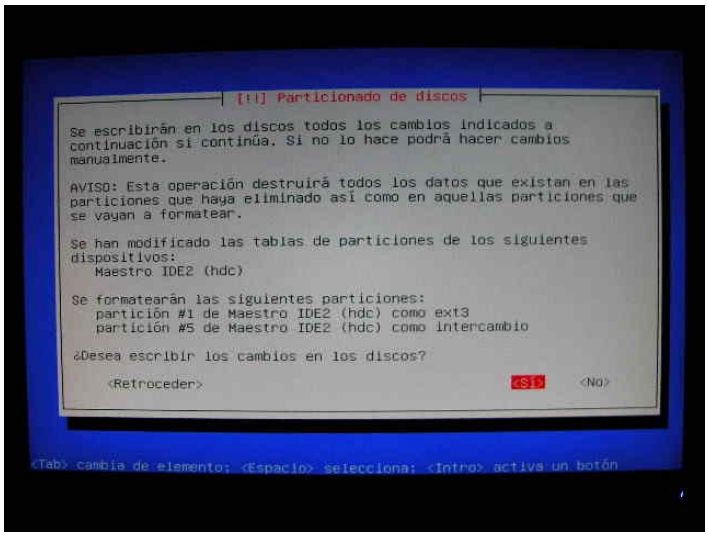
Ahora nos deja elegir entre tres opciones, para simplificar un poco las cosas podemos seleccionar el modo (*recomendado para novatos*) que tan solo nos creara una partición **raiz /** y una **swap**. Podemos separar la partición **/home** de la **/** ademas de muchas otras mas pero esta vez vamos a instalar todos los ficheros en una partición y seguiremos adelante …



Si ves que todo esta correcto (**Finalizar el particionado y escribir los cambios en el disco**) …

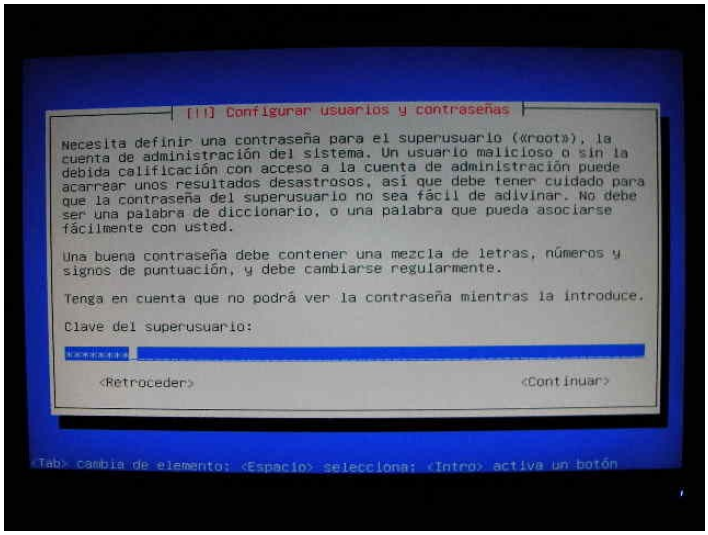


El sistema nos pide permiso para realizar los cambios en las particiones, echarle un vistazo y si todo esta correcto adelante !!

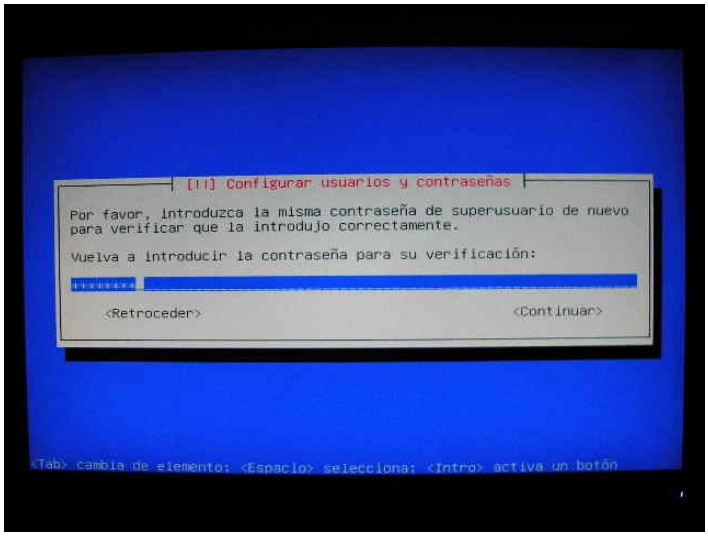


Selecciona tu zona horaria …

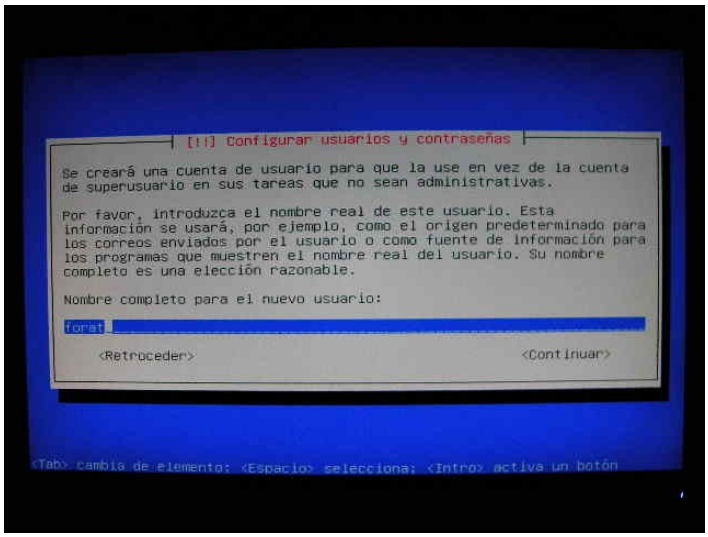
Ahora nos pide la clave de super usuario (**root**), os recomiendo que la apuntéis ya que la necesitaremos mas adelante …



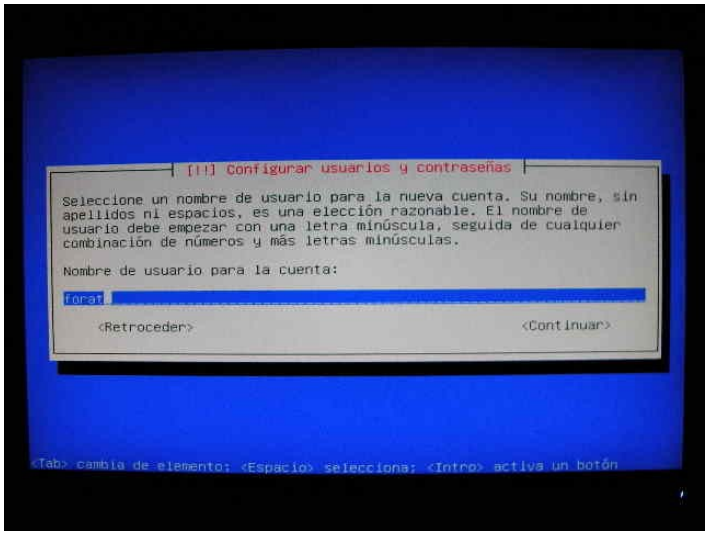
reescribe la contraseña …



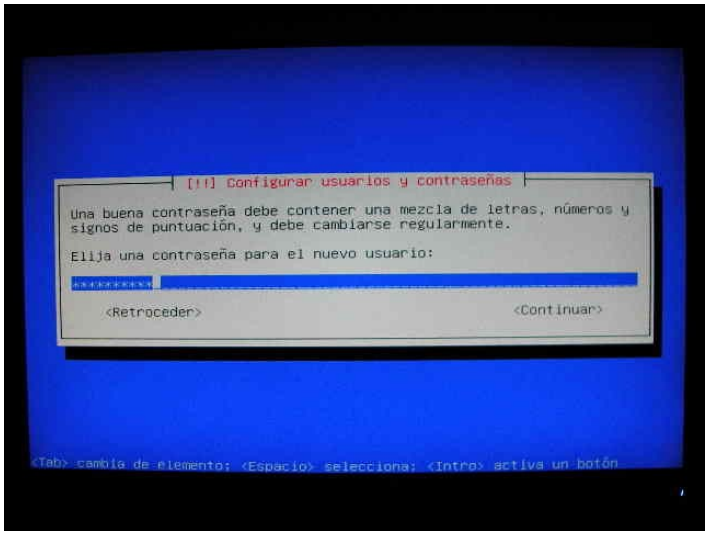
Ahora vamos a crear un usuario …



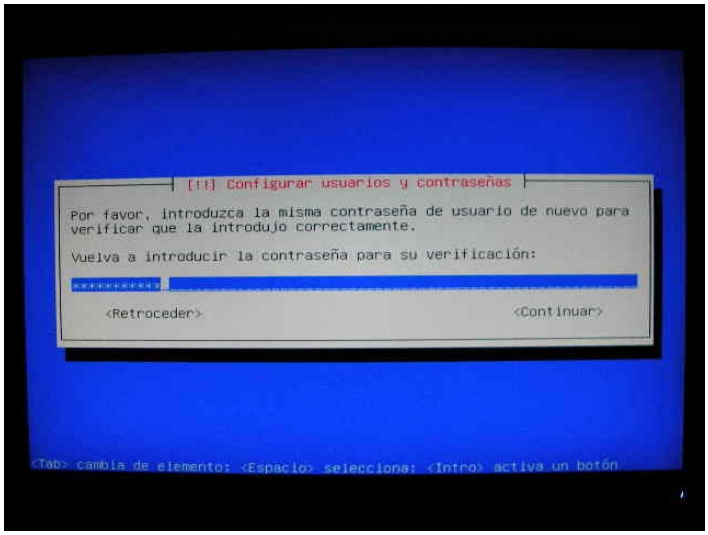
Podemos dejar el mismo …



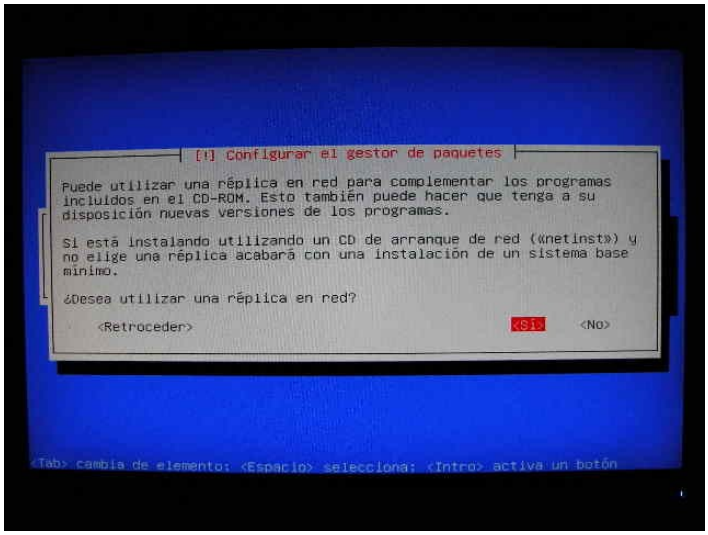
introduce una contraseña para el usuario …



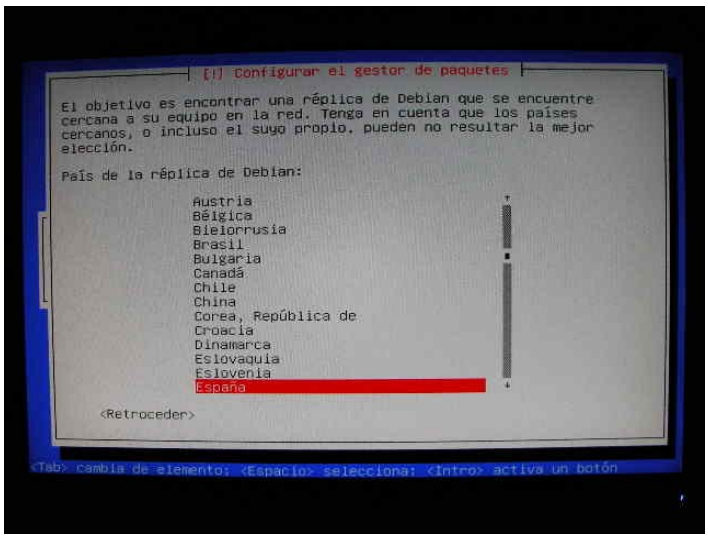
reescribe la contraseña …



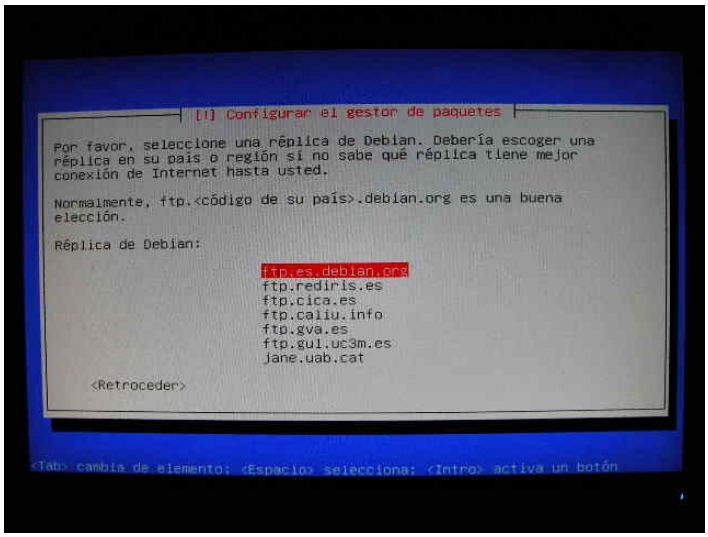
Si ya sois usuario de **Linux Ubuntu** seguro que esto os suena, se refiere a los repositorios. Los necesitamos para tener los programas actualizados e instalar programas …



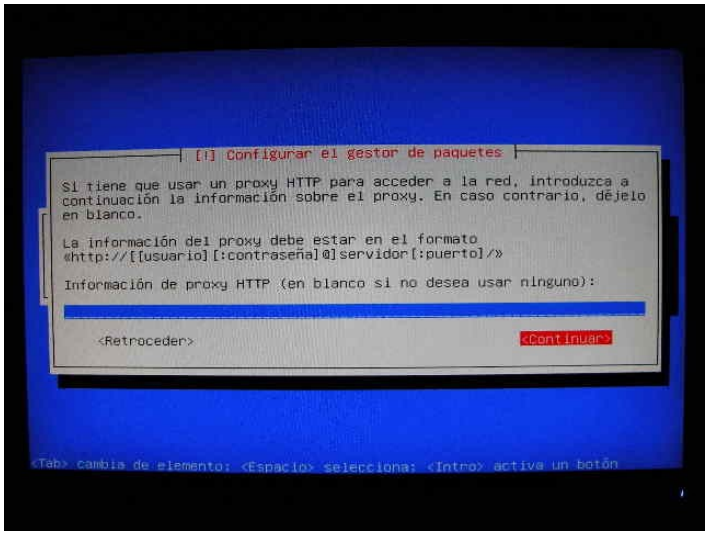
Elegimos el país que queremos usar para los repositorios …



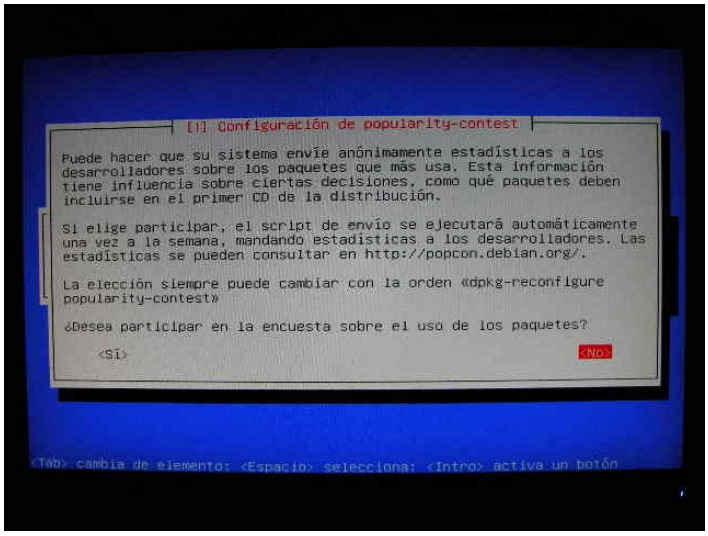
Elegimos la replica …



No usamos proxy, continuamos …

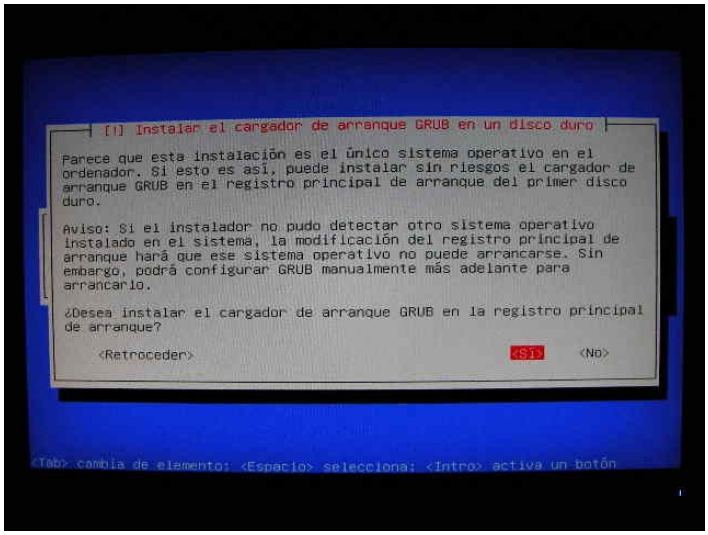


Esto es opcional …

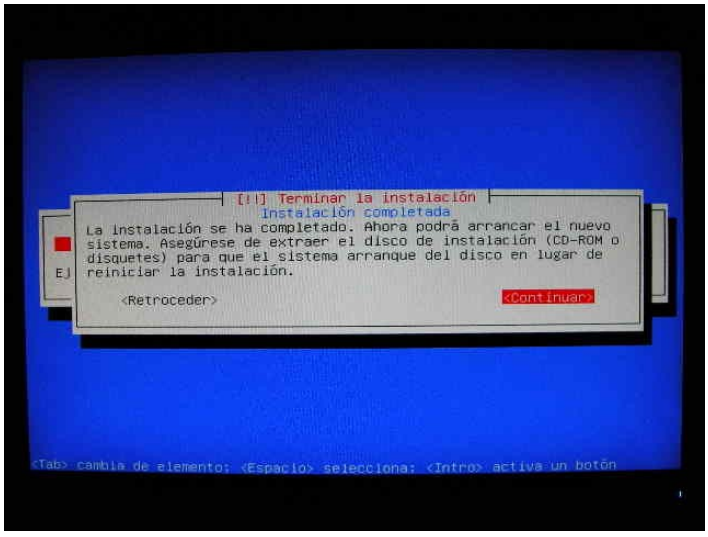


En el siguiente apartado desmarcamos todas las opciones menos la de (**Sistema estándar**)

Nos pregunta si queremos instalar **Grub** y le decimos que si …



Instalación completada !!! ahora se abrirá la compuerta del CD y listo …



Sacamos el cd y reiniciamos …

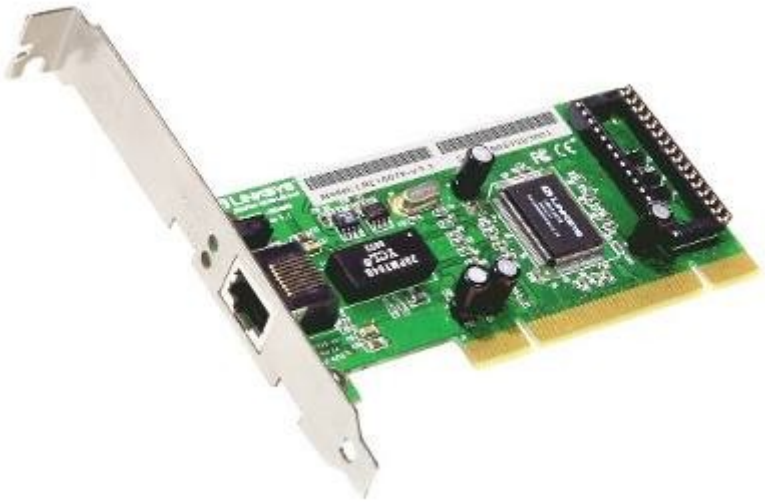
A partir de aquí ya tenemos el sistema operativo **Linux Debian** correctamente instalado, en el siguiente capitulo de este manual veremos como configurarlo a conciencia para que podamos usarlo como **servidor**.

A por cierto he aprovechado el montaje de este servidor para incluir la primera tarjeta gráfica que tuve en mi vida, era una Magic Combo CGA monocromo y un monitor de 13 pulgadas CGA monocromo que restaure pintando la carcasa exterior de negro

Aquí tenemos una muestra de lo poco necesario que son los recursos gráficos en este servidor ya que esta tarjeta y este monitor no soportan elementos gráficos. Bueno si en código **ASCII**.

Pensaba incluir en este articulo todas las configuraciones que vamos a darle al sistema pero voy ha hacerlo en varios artículos porque si no esto seria interminable.

Configuración de red



Una de las primeras cosas que vamos ha hacer después de **instalar Linux Debian** en este servidor que nos estamos montando es configurar la red. Nos es completamente necesario tener una IP fija en este ordenador ya que la vamos a necesitar para poder asignarle unos cuantos puertos en el Router que son necesarios para que se puedan ver servicios desde Internet como por ejemplo el 80 para el Web o el 21 para el Ftp.

Una vez instalada correctamente la distribución arrancamos el ordenador y nos identificamos como (**root**) …

Vamos a hacer algunas comprobaciones y configuraciones para asegurarnos de que tenemos bien configurada la red además de tener acceso a Internet desde este servidor que nos estamos montando.

Para empezar vamos a asignarle una **IP fija** a nuestra maquina, esto es necesario porque si vamos a ofrecer servicios hacia Internet deberemos abrir puertos y asignarlos a una **IP fija** desde el **Router.** Si por el contrario tenemos la asignación **IP** de forma automática con **DHCP** no podremos asignar X puerto a X IP. Con lo cual es completamente necesario que nuestra tarjeta de red tenga un **IP fija**.

Si instalasteis **Linux Debian** como os indicaba en el articulo anterior tendréis que tener una **IP** asignada de modo automático por el **Router**. Comprobemos si tenemos salida a Internet mandando unos pings, por ejemplo a **Google** …

~ codigo ~

ping www.google.com

Salimos del ping con (**Control + c**)

Si tenemos respuesta tenemos **IP**, ahora vamos a ver de que rango es la **IP** que nuestro **Router** nos asigno en el arranque ejecutando el comando **ifconfig** …

~ codigo ~

ifconfig

Veremos algo parecido a esto …

Como podemos ver en la imagen, mi Router tiene activado el modo **DHCP** y me asigno automaticamente la IP **192.168.1.64** por lo que con ella sabemos el rango que debemos usar para nuestra **IP fija** es de **192.168.1.x**.

Antes de nada nos hacemos una copia de seguridad de el archivo (**interfaces**), si algo falla siempre lo tendremos hay …

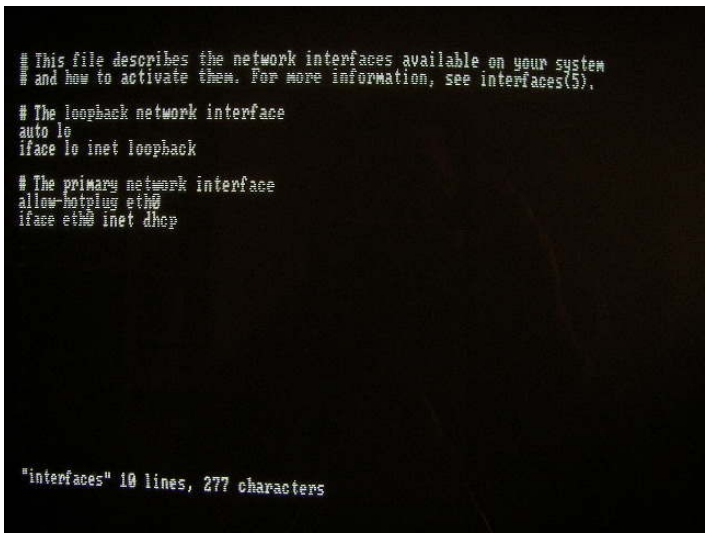
- codigo -

cp /etc/network/interfaces /etc/network/interfacesOK

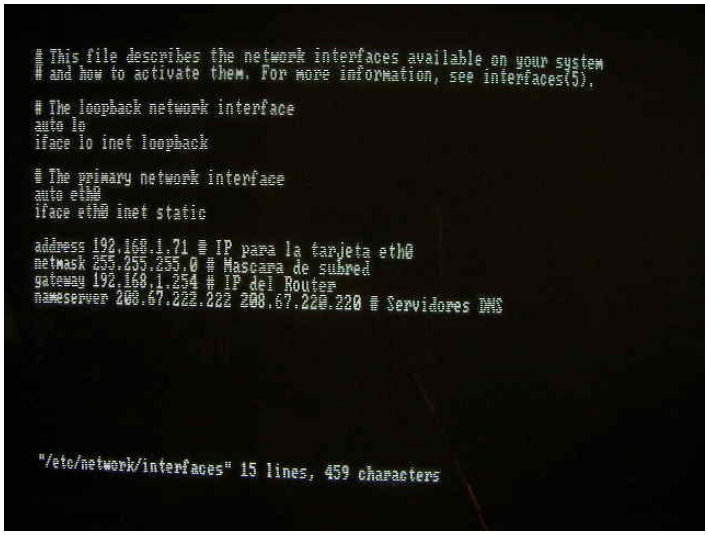
Ahora editamos el archivo **/etc/network/interfaces** para cambiar la configuración de nuestra red…

- codigo -

vi /etc/network/interfaces



Como podéis ver a nuestra tarjeta de red se le asigna una **IP automática** gracias a la linea (**iface eth0 inet dhcp**). A continuación vamos a introducirle los siguientes parámetros para **configurar manualmente la tarjeta de red** …



Si nos fijamos en la imagen podemos ver la dirección de mi tarjeta de red, mascara de subred, IP del Router y servidores de nombres DNS. Substituir mis parámetros por los vuestros. Una vez terminados los cambios pulsamos la tecla (**ESC**) , guardamos y salimos con (**:wq!**)

Ahora vamos ha hacer unas pruebas para ver si lo hemos hecho bien. Reiniciamos la maquina con …

- codigo -

reboot

Una vez reiniciada la maquina nos logeamos en el sistema como usuario (**root**) …

Seguidamente le hacemos un **Ping** a **Google** para ver si tenemos acceso a Internet …

~ codigo ~

ping www.google.com

Por lo visto tenemos conexión !! Ahora veamos la configuración de la tarjeta con el comando (**ifconfig**) …

Como podéis ver la dirección IP de la tarjeta de red ha cambiado por la que le asignamos. Ahora cuando necesitemos redireccionar un puerto desde el **Router** a nuestra maquina nos será posible ya que tenemos una **IP fija** (*en mi caso la* **192.168.1.71**). He puesto esta IP porque la **192.168.1.70** es de mi servidor **hsnet.sytes.net**, pongo las IPs correlativas para no hacerme un lío de maquinas. (*Tengo unas cuantas*) Si es tu caso toma nota

Respecto al **Router** y como abrir o cerrar puertos o mejor dicho redireccionar puertos hablaremos en otro articulo cuando le llegue su turno de momento ya tenemos una cosa mas hecha en nuestro servidor además de haber aprendido como es la **configuración de red en Linux Debian**.

Secure Shell (SSH)

¿ Seria bonito poder apagar el monitor de nuestro servidor y administrarlo desde una Terminal en nuestro escritorio verdad ?

Esto lo vamos ha poder hacer prácticamente desde cualquier sistema operativo, así que si tu ordenador de escritorio también llamado (**WorkStation**) no tiene **Linux** no te preocupes que también podrás hacerlo.

(***Hay que pensar en todos***)

SSH (Secure SHell) es una forma segura con la que vamos a poder conectarnos a nuestro servidor de manera remota. Podremos conectarnos a el por nuestra red siendo de manera local o desde Internet. De momento nos conformaremos con poder entrar desde otro PC que tengamos en nuestra red ya que todavía no os he explicado como vamos a localizar nuestro servidor desde Internet, todo se andará.

Arrancamos nuestro servidor y nos identificamos como (**root**), seguidamente vamos a proceder a actualizar la información de los repositorios ya que si estáis siguiendo al pie de la letra el manual no lo hemos hecho aun, así que ejecutamos los comandos …

~ codigo ~

apt-get update

Seguidamente instalamos el programa para el acceso remoto seguro **SSH** …

~ codigo ~

apt-get install ssh

Ahora reiniciamos la maquina para ver si arranca bien en el inicio …

~ codigo ~

reboot

Una vez tengamos reiniciada la maquina vamos a conectarnos a ella desde otro ordenador de nuestra red. Si recordamos el articulo anterior configuramos una IP fija para nuestra tarjeta de red, así que ahora nos vamos a conectar a esa IP.

Para los que tenemos Linux nos va a ser fácil conectarnos ya que no tenemos que instalar ningún programa (***las Terminales vienen de serie en Linux***) así que abrimos una terminal y ejecutamos …

~ codigo ~

ssh 192.168.1.70

Veremos algo parecido a esto …

El sistema **Secure SHell** genera una llave de autentificación y si nos equivocamos el sistema nos beta para no dejarnos pasar mas. Esta es una de las medidas de seguridad que nos ofrece este programa.

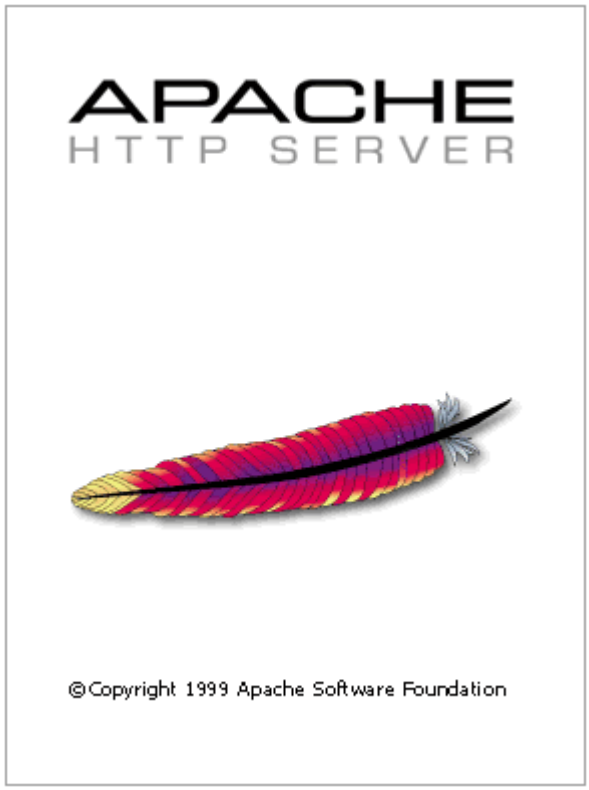
Le indicamos **YES** …

Como podéis ver ya estamos dentro de nuestro servidor desde una terminal en nuestro ordenador que usamos habitualmente. Con esto hemos conseguido poder abrirnos una puerta para poder entrar en nuestro servidor bajo modo local y mas adelante os enseñare como conectarnos a el desde Internet.

Para conectarme con el servidor yo estaba identificado en mi ordenador portátil como el usuario **forat**, el sistema detecta el usuario en el que estas identificado automáticamente y pide el password de el mismo que se encuentra creado en el servidor. Por lo que si entramos como usuario y queremos identificarnos como administrador del sistema (**root**) tan solo tenemos que insertar el comando **su** e introducirle el password de (**root**).

Para los que necesitéis en un momento dado administrar vuestro servidor desde una Terminal en vuestro sistema y no tengáis Linux, esto no es un problema ya que hay varias utilidades para esto. Para **Windows** podéis usar el archi-famoso cliente llamado **Putty** y para MAC podéis usar el cliente para terminales **MacSSH**.

Apache Web Server



Vamos a cumplir una de las finalidades de este gran manual que estamos siguiendo últimamente y es el de poder servir contenido Web desde nuestro servidor hacia nuestra red local. Digo esto porque hacia Internet podremos hacerlo mas adelante, vallamos por orden para que todo sea mas comprensible.

Para esto usaremos el programa **Apache Web Server** en su versión **apache2-mpm-prefork** que es la recomendada para **Debian** por su **Multi-Processing Module (MPM)** ya que es considerablemente más rápido que el modelo tradicional. Si deseas mas información sobre el tema del (MPM) **aquí os dejo una explicación**.

Procedemos a arrancar nuestro servidor y nos logeanos como (**root**).

Seguidamente instalaremos el paquete **Apache2** y acto seguido el **apache2-mpm-prefork** …

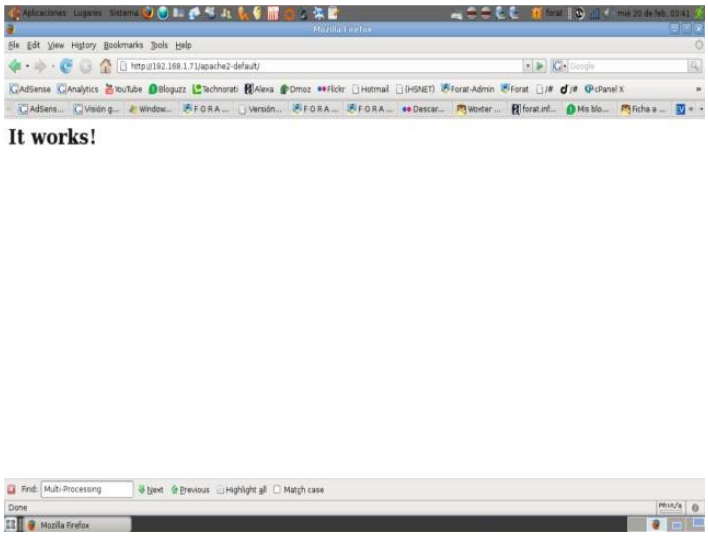
~ codigo ~

apt-get install apache2

apt-get install apache2-mpm-prefork

Una vez finalizada la instalación abrimos un navegador en nuestro PC de escritorio e introducimos la IP del servidor (**en mi caso http://192.168.1.71**) y

veremos el mensaje (**It works!**), algo así …



Este mensaje quiere decir que el *servidor Web Apache está funcionando* y listo para servir contenido Web.

El directorio hacia donde apunta el **servidor Web Apache** por defecto es a **/var/www/apache2-default**. Esto lo podemos cambiar para que apunte a uno de los directorios que tenemos en nuestro directorio **/home/usuario** para que cuando mas adelante instalemos el servidor **Ftp** podemos subir y bajar archivos directamente desde un directorio que será visto por los demás desde Internet o desde nuestra área local.

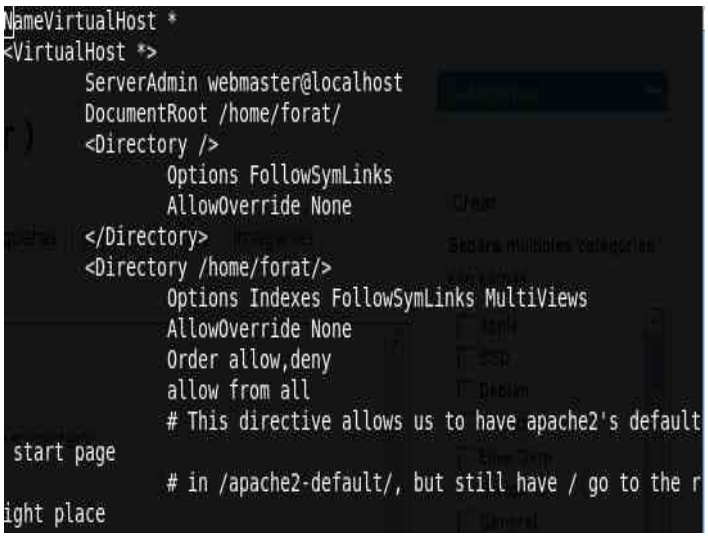
Para hacer esto vamos a editar el archivo **default** que se encuentra en la ruta **/etc/apache2/sites-available** y vamos a substituir la ruta **/var/www/** por la nuestra de nuestro **/home/usuario/** …

Editamos el fichero con el editor **vi** …

~ codigo ~

vi /etc/apache2/sites-available/default

y modificamos los directorios …



```
NameVirtualHost *
<VirtualHost *>
        ServerAdmin webmaster@localhost
        DocumentRoot /home/forat/
        <Directory />
                Options FollowSymLinks
                AllowOverride None
        </Directory>
        <Directory /home/forat/>
                Options Indexes FollowSymLinks MultiViews
                AllowOverride None
                Order allow,deny
                allow from all
                # This directive allows us to have apache2's default
 start page
                # in /apache2-default/, but still have / go to the r
ight place
```

También he modificado la linea …

RedirectMatch ^/$ /apache2-default/

por

RedirectMatch ^/$ /www/

Esto a sido algo personal y de gusto, particularmente prefiero que el directorio donde se va a almacenar el contenido Web que quiero ofrecer a los usuarios tenga por nombre (**www**) que (**apache2-default**).

Ahora vamos a crear el directorio hacia donde va a apuntar el servidor **Web** …

~ codigo ~

mkdir /home/forat/www

y reiniciamos el **Apache** con …

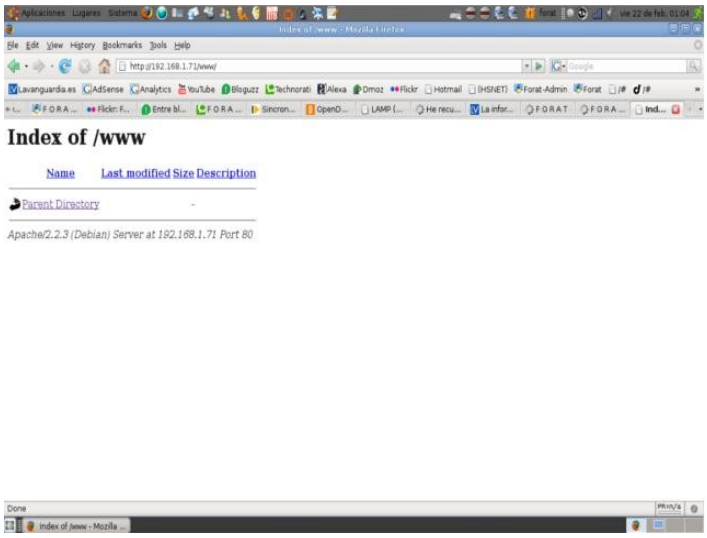
~ codigo ~

apache2ctl restart

Ahora volvemos a abrir un navegador en el ordenador de escritorio e introducimos la dirección **IP** del servidor.

(*En mi caso* **http://192.168.1.71**)

Veremos algo como esto …



Como podéis ver todo lo que copiemos a la carpeta **www** dentro de nuestro directorio **/home/usuario** se podrá ver desde otros navegadores ..

Soporte para PHP

Uno de los lenguajes de programación que vamos a necesitar en el **montaje de este servidor Web en Linux Debian** es sin duda (**PHP**). Este lenguaje es usado por el software de **WordPress** con el que al final de este gran manual nos montaremos un Blog.

Vamos a instalar unos cuantos paquetes para que nuestro servidor sea compatible con **PHP** y podamos usarlo tanto programar *webs*, *scripts* o *cgis* del siguiente modo …

Arrancamos nuestro servidor y nos logeamos como (**root**) e instalamos los paquetes **php4-cgi, php4-cli, php4-common y libapache2-mod-php4**.

- codigo -

apt-get install php4-cgi php4-cli php4-common libapache2-mod-php4

Vamos a agregar la siguiente linea (**DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.shtml**) al fichero de configuración del **Apache** llamado **apache2.conf** para que nuestro Web server sea compatible con ese tipo de extensiones …

- codigo -

echo “DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.shtml” >> /etc/apache2/apache2.conf

Podéis verificar si lo habéis hecho bien con el comando (**cat**) que nos muestra

en pantalla el contenido de un archivo …

- codigo -

cat /etc/apache2/apache2.conf

Tenemos que ver la ultima linea como esta …

Ahora vamos a comprobar que todo ha ido bien editando un fichero en el directorio raíz de nuestro servidor Web. Si recordáis el manual anterior (**Apache Web Server**) configuramos a conciencia donde iba a estar situada la raíz de el servidor Web, osea el directorio que es visible desde otros exploradores, (**en mi caso /home/forat/www/**).

Creamos un archivo llamado por ejemplo … **compatible.php** con …

- codigo -

vi /home/usuario/www/compatible.php

e insertamos el siguiente código …

```
<?php  phpinfo();  ?>
```

Guardamos el archivo **compatible.php** y cerramos el editor pulsando la tecla (**ESC**) y (**:wq!**).

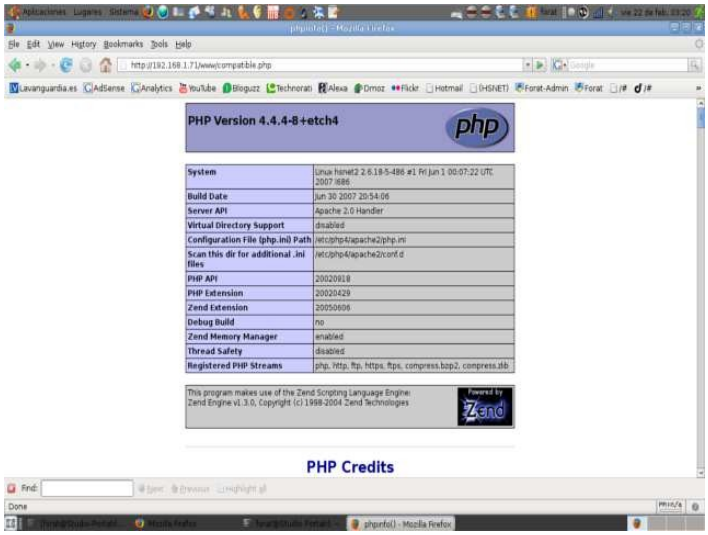
Una vez completados estos pasos vamos a reiniciar el **Apache2** con …

~ codigo ~

apache2ctl restart

Ahora solo tienes que abrirlo en un navegador y insertar la dirección de tu servidor Web (*en mi caso* **http://192.168.1.71/www/compatible.php**)

Veremos información detallada sobre el soporte para PHP que ahora tenemos instalado …



Ya estamos listos para continuar con el siguiente capitulo ...

Bases de datos MySQL

En el siguiente articulo vamos a montar **MySQL** y **PhpMyAdmin** para poder administrar las bases de datos que mas adelante nos aran falta para crear un **Blog** con **Wordpress**.

Por si no lo sabíais todos los **Blogs** que podemos ver en la red están vinculados a una base de datos que pueden estar en otros lenguajes pero nosotros vamos a instalar las **MySQL**. Hace tiempo que trato con ellas y son bastante fiables, manejables y rápidas para interactuar con la Web.

Vamos a instalar unos cuantos paquetes para que nuestro servidor pueda ofrecernos bases de datos **MySQL** del siguiente modo …

Arrancamos nuestro **servidor en Linux Debian** y nos identificamos como (**root**) e instalamos los siguientes paquetes …. (*Ten el CD de* **Debian** *a mano que os lo pedirá seguramente*)

~ codigo ~

apt-get install mysql-server-4.1 mysql-client-4.1 php4-mysql

Para facilitarnos manejo y la administración de nuestras bases de datos vamos a usar el entorno web **PhpMyAdmin**, con este software podremos crear, borrar, modificar, dar permisos, bueno podemos hacer todo a lo que se refiere a la administración de bases de datos **MySQL.** Comenzamos la instalación con …

~ codigo ~

apt-get install phpmyadmin

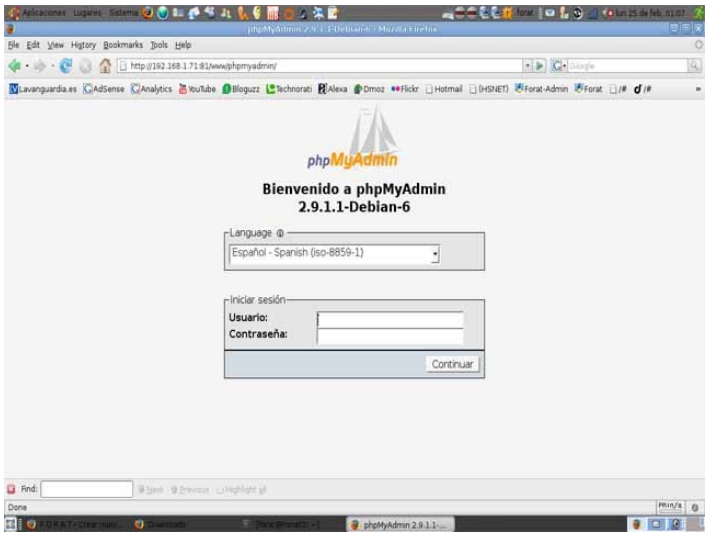
Seguramente recordareis el directorio raíz de nuestro servidor web. Resulta que el programa **phpmyadmin** se instala en la ruta **/var/www/** de nuestro **apache2** por defecto pero nosotros la vamos a enlazar a **/home/usuario/www/** para poder acceder a ella desde un navegador, así que ahora tendremos que hacer otro enlace simbólico para poder ver el entorno web de **phpmyadmin** de la siguiente manera …

~ codigo ~

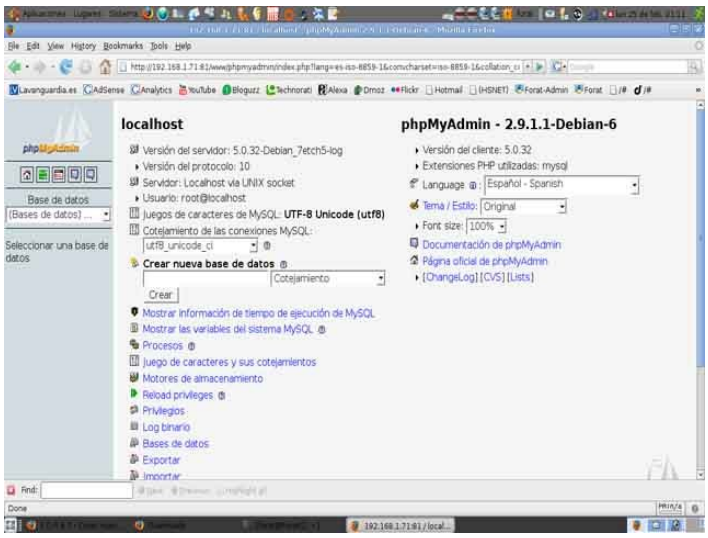
ln -s /var/www/phpmyadmin /home/usuario/www/phpmyadmin

Una vez creado el enlace debemos ver ya el directorio de la aplicación en la raíz de nuestro servidor. Abrimos un navegador e introducimos la dirección **http://ipdelservidor/www/phpmyadmin/** (*En mi caso* **http://192.168.1.71/www/phpmyadmin/**)como podéis ver en la imagen …

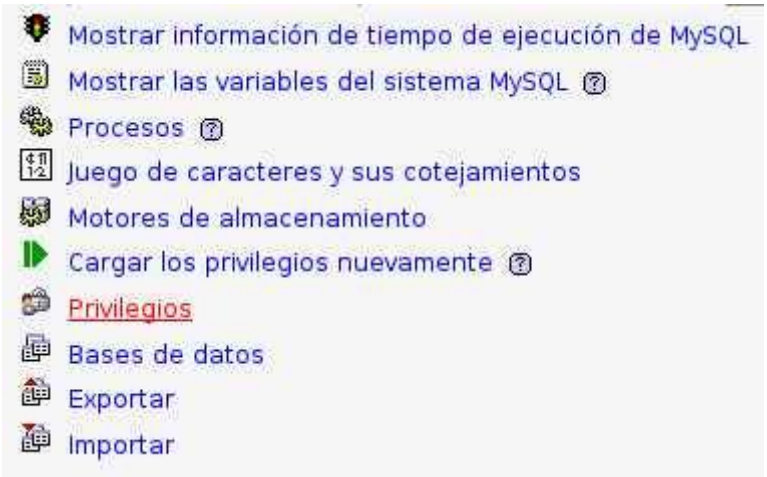
Pincha sobre el directorio de **phpmyadmin** y veras una pantalla como esta …



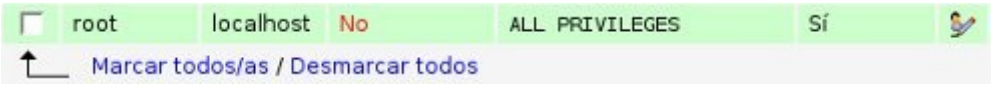
Seguidamente introducimos en la casilla de *usuario* (**root**), la de *password* la dejamos en blanco y pulsamos la tecla (**Enter**) …



Antes de nada vamos a eliminar esta brecha de seguridad que por defecto nos ofrece este programa. De momento podemos administrar las bases de datos sin **password**, esto mismo es lo que vamos a cambiar accediendo al apartado de privilegios …



A continuación editaremos el usuario **root** para **localhost** pinchando sobre el icono (**Editar**)



Y buscamos la sección de cambio de contraseña que se encuentra mas abajo …

Introducimos el nuevo **password** en las dos casillas y pinchamos en el botón (**Continuar**)

Ahora ya lo tenemos todo listo y seguro. Recordad que a partir de ahora necesitaremos ese **password** para administrar las bases de datos que mas adelante crearemos.

Recordad apuntaros todos los **passwords** que estamos usando para crear este servidor ya que os irán haciendo falta con el tiempo y el uso que le demos a nuestro invento casero

De momento hemos creado en todos los capítulos los usuarios …

Usuario normal, en mi caso (**forat**)

- - *Usuario administrador del sistema*, en mi caso y en el de todos vosotros (**root**)

- *Usuario administrador de las bases de datos MySQL* (**root**)

Ya tenemos soporte para bases de datos SQL ...

Información del sistema

con PhpSysInfo

PhpSysInfo es un **script** escrito en **PHP** que analiza todos los recursos que nuestro servidor nos ofrece respecto al **Hardware**. Con este software podremos ver el tiempo que lleva activo nuestro servidor y una descripción exacta del **Hardware** que tenemos un nuestra maquina como es el microprocesador, memoria, discos duros, etc …

Una vez lo tengamos instalado podremos ver que tal vamos de recursos del sistema como por ejemplo el uso del disco duro y el uso de memoria, estos son importantes para tener una idea de lo que vamos necesitando en nuestro servidor.

La instalación de este es muy sencilla, arrancamos nuestro **servidor en Linux Debian** y nos logeamos como (**root**) e insertamos …

~ codigo ~

apt-get install phpsysinfo

Seguramente recordareis el directorio raíz de nuestro servidor web.

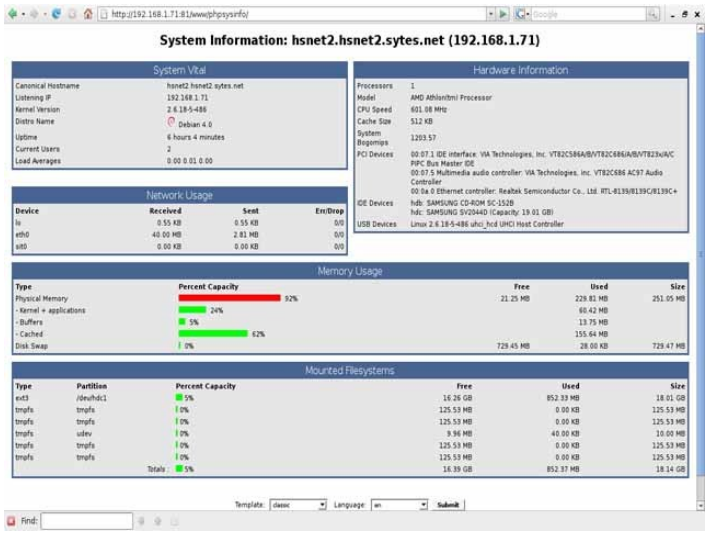
Resulta que el programa **phpsysinfo** se instala en la ruta **/var/www/** de nuestro

apache2 por defecto pero nosotros la vamos a enlazar a **/home/usuario/www/** para poder acceder a ella desde un navegador, así que ahora tendremos que hacer otro enlace simbólico para poder ver el entorno web de **phpsysinfo** de la siguiente manera …

~ codigo ~

ln -s /var/www/phpsysinfo /home/usuario/www/phpsysinfo

Una vez creado el enlace debemos ver el directorio de la aplicación en la raíz de nuestro servidor. Abrimos un navegador e introducimos la dirección **http://ipdelservidor/www/phpsysinfo/** (*En mi caso* **http://192.168.1.71/www/phpsysinfo/**)como podéis ver en la imagen …



Como podemos ver en la imagen nos muestra todo tipo de información sobre nuestra maquina. *Chulo verdad ;)*

Ahora vamos a editar el fichero **/etc/phpsysinfo/config.php** para que nos muestre el contenido en español o en el idioma que queramos por defecto …

~ codigo ~

vi /etc/phpsysinfo/config.php

```
<?php

$sensor_program = "";
$show_mount_point = false;
$show_bind = false;
$default_lng = "en";
$default_template = "classic";
$loadbar = false;
$hide_picklist = false;
$hddtemp_avail = "none";

?>
```

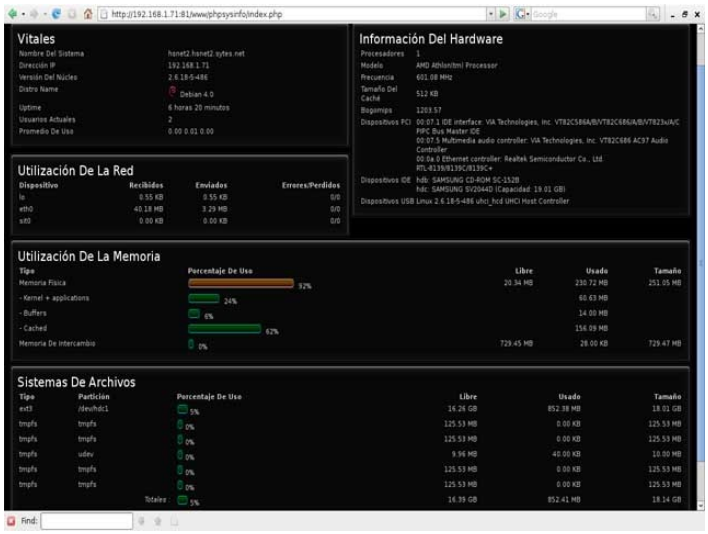
Si modificamos la linea **$default_lng = “en”** por **$default_lng = “es”** nos mostrará el contenido en español. En este apartado podemos modificar varias cosas interesantes como puede ser el tema por defecto que se mostrara cada vez que se visualicen las estadísticas, en mi caso he modificado varias. *Aquí os dejo una captura de mi archivo de configuración* **config.php** …

```
<?php

$sensor_program = "";
$show_mount_point = false;
$show_bind = false;
$default_lng = "es";
$default_template = "black";
$loadbar = false;
$hide_picklist = none;
$hddtemp_avail = "none";

?>
```

Aquí podéis ver como me a quedado a mi …



Una vez terminadas las modificaciones salimos del editor vi con la tecla (**ESC**) y (**:wq!**).

El otro servidor que tengo en casa va algo bastante mas sobrado que este, he aquí la diferencia de tener mas maquina o menos. Podeis ver las estadísticas OnLine **desde aquí**. Mas adelante también podréis ver las vuestras desde Internet (*Todo a su tiempo*)

A todo administrador de sistemas le gusta observar sus estadísticas y no podemos descuidar la de nuestro **Hardware** ya que si nuestro servidor se queda sin espacio en disco o nos quedamos sin memoria o tenemos algún fallo de **Hardware** podremos verlo reflejado aquí.

Por cierto para los que tenéis mas de 256 Mb de memoria no os preocupéis si veis la barra de estado de la memoria en rojo continuamente ya que de momento no hay problema con las aplicaciones que le hemos cargado. Para los que tenéis menos de 256 Mb de memoria iréis viendo que conforme usemos y metamos mas cosas en el servidor veremos la barra de estado al rojo prácticamente siempre, si veis que el servidor tarda mucho en responder es posible que tengáis que pensar en hacer una ampliación de memoria.

De momento tranquilos podéis continuar sin problemas, es posible que tengáis problemas cuando lo conectemos a Internet y empiecen a entrar los usuarios.

File Transfer Protocol
un FTP con ProFtpd

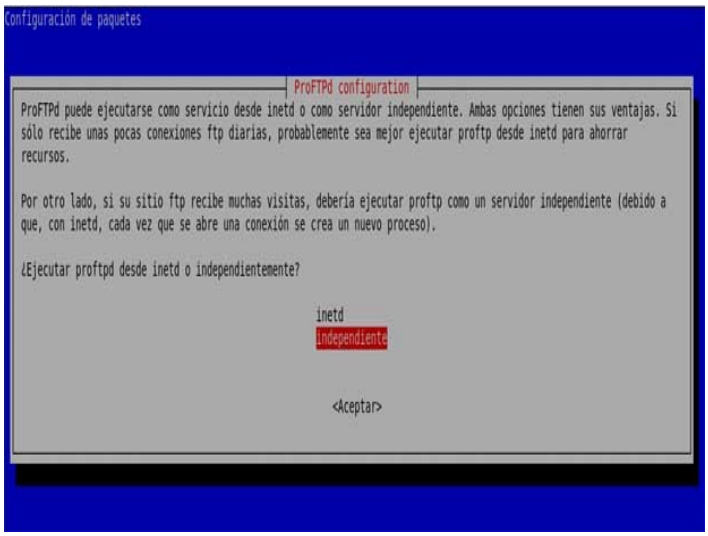
Se que mas de uno estaba esperando a que apareciera el manual para montar el FTP, una vez completemos este basico manual podremos subir y bajar archivos remotamente. Para esto vamos a usar el software (**Proftpd**) y configuraremos algo de el, aunque seguramente mas adelante publicare una configuración mas personalizada reforzando la seguridad.

Empezaremos la instalación de **Proftpd** arrancando nuestro **servidor en Linux Debian**, nos logeandonos como (**root**) y pasamos a instalar …

~ codigo ~

apt-get install proftpd

Durante la instalación debemos elegir entre las dos opciones que nos marca ….



Seleccionaremos (**Independiente**) y pulsamos la tecla (**Enter**)

A continuación vamos a agregar la linea **DefaultRoot ~** en el archivo de configuración **proftpd.conf** con el comando **echo** …

~ codigo ~

echo DefaultRoot ~ >> /etc/proftpd/proftpd.conf

Con esto conseguimos que no se pueda subir a mas directorios que el asignado para el usuario (**/home/usuario/**) Si no introducimos esta linea cualquiera que se conecte al servidor **FTP** podrá subir por los directorios y ver una información que se supone que no debe ver.

Nota : *No borrar los archivos que ya hay dentro de los directorios ya que podemos borrar configuraciones de usuario de otros programas.*

Una vez hecho esto reiniciamos el demonio **Proftpd** con …

~ codigo ~

/etc/init.d/proftpd restart

Ahora podemos hacer la prueba y conectarnos desde nuestro **cliente FTP** favorito al servidor **FTP** que ahora tiene instalado nuestro servidor. Para acceder al servidor de momento usaremos la IP local del server. (*En mi caso* **192.168.1.71**).

Recordad que todo lo que incluyamos dentro del directorio (**www**) se podrá ver vía web desde la raíz del servidor. (*En mi caso* **http://192.168.1.71/www/**)

Para finalizar el manual quisiera incluir un apunte que el amigo **Pana** nos incluye en el manual ….

Una vez hayamos realizado todos los pasos, en el root terminal (o consola de comandos logeado como root) debemos darle permisos al directorio asignado para el FTP (en este caso, "/home/usuario/www") de la siguiente manera:

~ codigo ~

chmod 777 /home/usuario/www

Si no lo hacemos, no podremos subir archivos a través del servidor http://FTP.

Si necesitáis un programa **cliente FTP** rápido y funcional para **Linux Ubuntu** os recomiendo el **gFTP**, este es el que uso yo y me va de perlas.

Estadísticas Web con Awstats

Desde hace tiempo vengo usando en diferentes servidores Web que he montado los programas para estadísticas **AwStats** y **Webalizer**. Esta vez vamos a instalar **AwStats** ya que creo que es el mas completo de los que he probado y que nos podemos instalar en nuestro servidor gratuitamente.

Una vez completemos la instalación y configuración de este software tendremos información al detalle como por ejemplo puede ser el tiempo que ha estado un usuario conectado a nuestras webs, visitas por mes, semana, día, de donde provienen las visitas, en fin todo lo que necesitamos saber de lo que viene a ver que tenemos en nuestro servidor.

Para los Blogs que tengo fuera de mi servidor utilizo **Google Analytics** y he de decir que es súper completo pero para los servidores que tengo en casa me gusta usar uno interno dentro del propio servidor. Aquí podéis ver una **muestra de el AwStats en mi servidor actual**.

AwStats es un software mas comprensible y goza de un interfaz Web muy amigable podemos pasar al español.

Para su instalación vamos a comenzar por arrancar nuestro **servidor en Linux Debian** y nos identificamos como usuario (**root**), y lo instalamos con …

- codigo -

apt-get install awstats

Para su instalación en el sistema vamos a ejecutar un script de configuración escrito en **Perl** situado en la ruta (**/usr/share/doc/awstats/examples**) …

- codigo -

cd /usr/share/doc/awstats/examples
perl awstats_configure.pl

Para su configuración nos encontraremos con algunas preguntas que tendremos que responder. (**Lo que esta en negrita son las respuestas**)…

- And then, run configure.pl from this location.
Do you want to continue setup from this NON standard directory [yN] ? **Y**

- Enter full config file path of your Web server.
Example: /etc/httpd/httpd.conf
Example: /usr/local/apache2/conf/httpd.conf
Example: c:\Program files\apache group\apache\conf\httpd.conf
Config file path ('none' to skip web server setup):
> /etc/apache2/apache2.conf

- Check and complete web server config file '/etc/apache2/apache2.conf'
Add 'Alias /awstatsclasses "/usr/share/doc/awstats/examples/wwwroot/classes/"'
Add 'Alias /awstatscss "/usr/share/doc/awstats/examples/wwwroot/css/"'
Add 'Alias /awstatsicons "/usr/share/doc/awstats/examples/wwwroot/icon/"'
Add 'ScriptAlias /awstats/ "/usr/share/doc/awstats/examples/wwwroot/cgi-bin/"'
Add '' directive
AWStats directives added to Apache config file.

- Need to create a new config file ?
Do you want me to build a new AWStats config/profile file (required if first install) [y/N] ? **N**

- Restart Web server with '/etc/init.d/apache restart'

—–> Add update process inside a scheduler
Sorry, configure.pl does not support automatic add to cron yet.
You can do it manually by adding the following command to your cron:
/usr/share/doc/awstats/examples/wwwroot/cgi-bin/awstats.pl -update -config=myvirtualserver
Or if you have several config files and prefer having only one command:
/usr/share/doc/awstats/examples/tools/awstats_updateall.pl now
Press **ENTER** to continue…

- No config file was built. You can run this tool later to build as much config/profile files as you want.
Once you have a config/profile file, for example 'awstats.demo.conf',

You can manually update your statistics for 'demo' with command:
> perl awstats.pl -update -config=demo
You can also read your statistics for 'demo' with URL:
> http://localhost/awstats/awstats.pl?config=demo

Press **ENTER** *to finish…*

Una vez completada la instalación vamos a configurar el archivo (**awstats.conf**) que se encuentra en la ruta (**/etc/awstats/**) …

~ codigo ~

vi /etc/awstats/awstats.conf

Una vez dentro buscamos la linea …

LogFile="/var/log/apache/access.log"

y la substituimos por …

LogFile="/var/log/apache2/access.log"

Cambiamos la linea …

LogFormat=4

por …

LogFormat=1

Buscamos la linea …

SiteDomain=""'

Aquí vamos a indicarle cual es nuestro dominio, para esto vamos a usar el nombre que le pusimos a nuestro servidor (*en mi caso* **hsnet2**) y le agregaremos el subdominio **sytes.net**, aquí os dejo un ejemplo …

SiteDomain="hsnet2.sytes.net"

Una vez hechas estas modificaciones guardamos y salimos del editor **vi** con (**ESC**) y (**:wq!**).

Ahora vamos ha hacer una copia de este fichero pero con nuestro nombre de dominio … (*en mi caso* **hsnet2.sytes.net**) adjundando **awstats** y **conf** …

~ codigo ~

cp awstats.conf awstats.hsnet2.sytes.net.conf

Vamos a darle permisos al fichero **log** que almacena **apache2** para que **awstats** pueda leerlo sin problemas …

- codigo -

chmod o+r /var/log/apache2/access.log

Ahora vamos a ejecutar la configuración que antes hicimos con el archivo **awstats.conf** pero para nuestro servidor …

- codigo -

/usr/lib/cgi-bin/awstats.pl -config=awstats.hsnet2.sytes.net.conf

Una vez hecho esto se actualizara y tomara la información de los ficheros **Log** de nuestro **apache2**.

Ahora vamos a crear un enlace simbólico de la carpeta **cgi-bin** que **awstats** nos a creado en la ruta (**/usr/lib/**) hacia el directorio raíz de nuestro servidor Web con …

- codigo -

ln -s /usr/lib/cgi-bin /home/usuario/www/cgi-bin

Seguidamente configuraremos el fichero **apache2.conf** para que se muestren bien las imágenes en los resultados de las estadísticas …

- codigo -

vi /etc/apache2/apache2.conf

Nos situamos al final del fichero y modificamos las siguiente lineas …

Alias /awstatscss "/usr/share/doc/awstats/examples/wwwroot/css/"

por …

Alias /awstatscss "/usr/share/doc/awstats/examples/css/"

y …

Alias /awstatsicons "/usr/share/doc/awstats/examples/wwwroot/icon/"

por …

Alias /awstatsicons "/usr/share/awstats/icon/"

y …

ScriptAlias /awstats/ "/usr/share/doc/awstats/examples/wwwroot/cgi-bin/"

por …

ScriptAlias /awstats/ "/usr/lib/cgi-bin/"

También modificaremos …

por …

y finalmente agregaremos estas lineas al final del archivo …

```
Alias /awstats-icon/ /usr/share/awstats/icon/
<Directory /usr/share/awstats/icon>
  Options None
  AllowOverride None
  Order allow,deny
  Allow from all
</Directory>
```

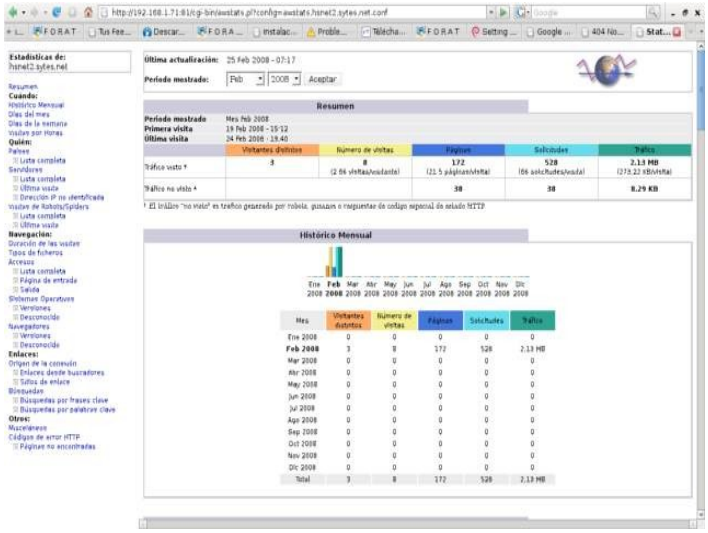
Una vez efectuados los cambios guardamos y salimos con la tecla (**ESC**) y (**:wq!**).

Ahora podemos abrir un navegador y veremos los resultados de todo esto

Cambia la IP por la IP de tu servidor Web y el dominio por tu dominio …

http://*192.168.1.71***/cgi-bin/awstats.pl?config=awstats.***hsnet2.sytes.net***.conf**

Como podéis ver el programa de estadísticas esta listo para mostrarnos todo lo que hacen nuestros usuarios, con que lo hacen y cuando lo hacen. Que bonito !!!



Ya tan solo nos queda un ultimo paso para automatizar la tarea de actualización de este programa. Para esto vamos a usar el programador de tareas (**Crontab**) que lleva nuestra **Linux Debian** de serie.

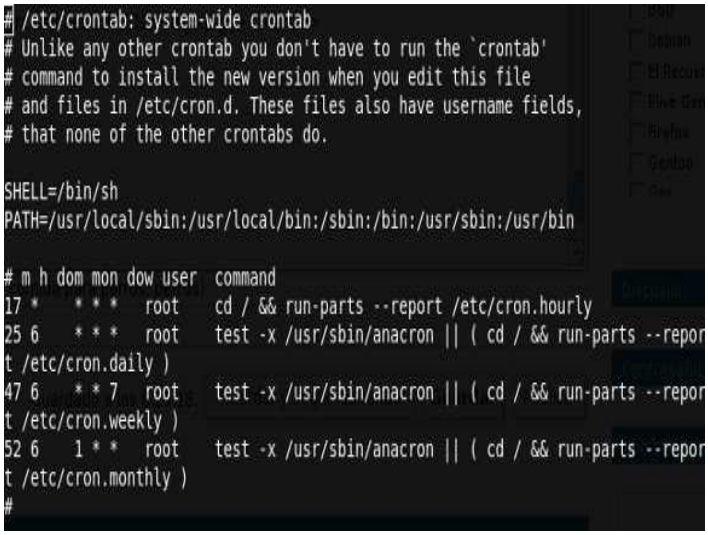
Con lo que veremos a continuación vamos ha hacer que se actualice una vez cada 60 minutos, así no sobrecargamos mucho nuestra maquina y tenemos prácticamente actualizada la información la hora de verla.

Programamos la tarea …

~ codigo ~

vi /etc/crontab

Veremos algo así …



Nos situamos en el final del fichero y pegamos esta linea …

0-59/60 * * * * root /usr/lib/cgi-bin/awstats.pl -config=hsnet2.sytes.net -update >/dev/null

Recordad modificar (**hsnet2** por vuestro servidor)

Ahora ya tenemos el programa **AwStats** funcionando y esperando movimiento en nuestro servidor para ofrecernos las estadísticas al detalle.

(**¡¡ Que para eso somos los administradores del sistema !!**)

Redireccionamiento DNS de No-IP

Para los que tenemos una **ADSL** normalita solemos tener por defecto **IP dinámica** y con ello un problema a la hora de localizar nuestro **Router ADSL** desde **Internet**.

De este modo nos va a ser imposible localizar nuestro servidor Web desde **Internet** ya que cada X's tiempo va cambiando nuestra dirección **IP**. Para poder localizar nuestro servidor desde Internet vamos a usar el servicio gratuito de redireccionamiento de **DNS** que nos ofrece la empresa **No-IP**.

De lo que se trata es de darnos de alta en la web de **No-IP** y elegir un dominio … Los que habéis seguido estos manuales sobre **como montar un servidor Web en Linux Debian** al pie de la letra seguro que os suena la dirección …. (**hsnet2.sytes.net**)

Exacto, lo que vamos a registrar es un **dominio.sytes.net** y en dominio pondremos el nombre de nuestro servidor como hemos hecho anteriormente en algunos artículos.

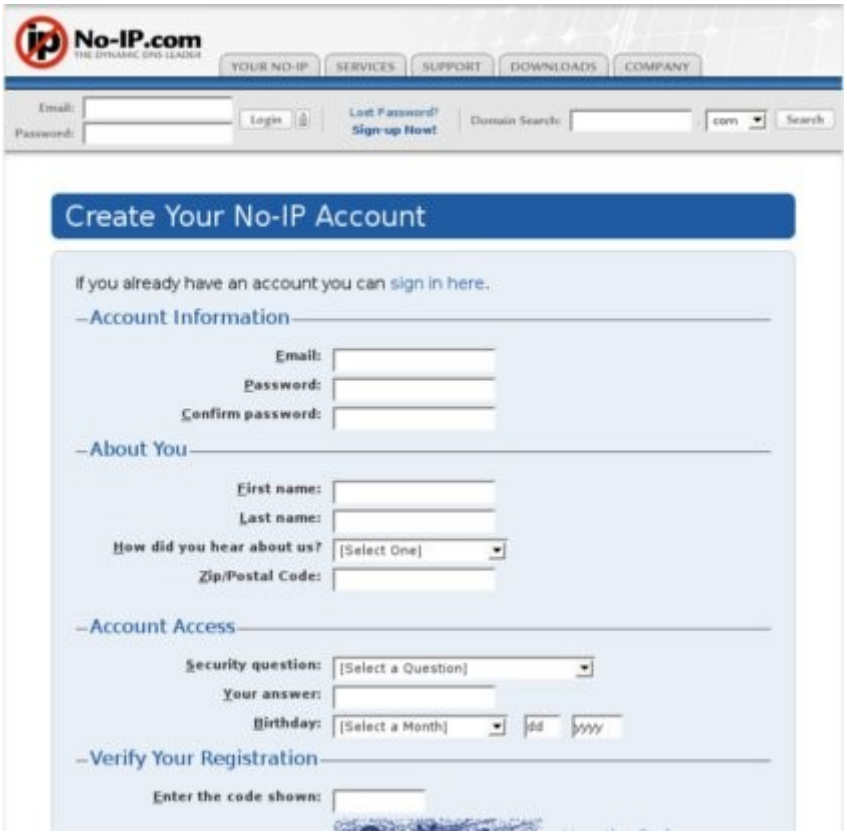
Una vez registremos el dominio instalaremos un programa en nuestro servidor que se encargara de mandar un pings cada 30 minutos para que cada vez que nos cambie nuestra **IP** sea relacionada con el dominio que registremos.

Esta es una manera eficaz de localizar tu servidor Web hospedado en casa detrás de tu **Router ADSL**.

¿ **Bonito verdad** ?

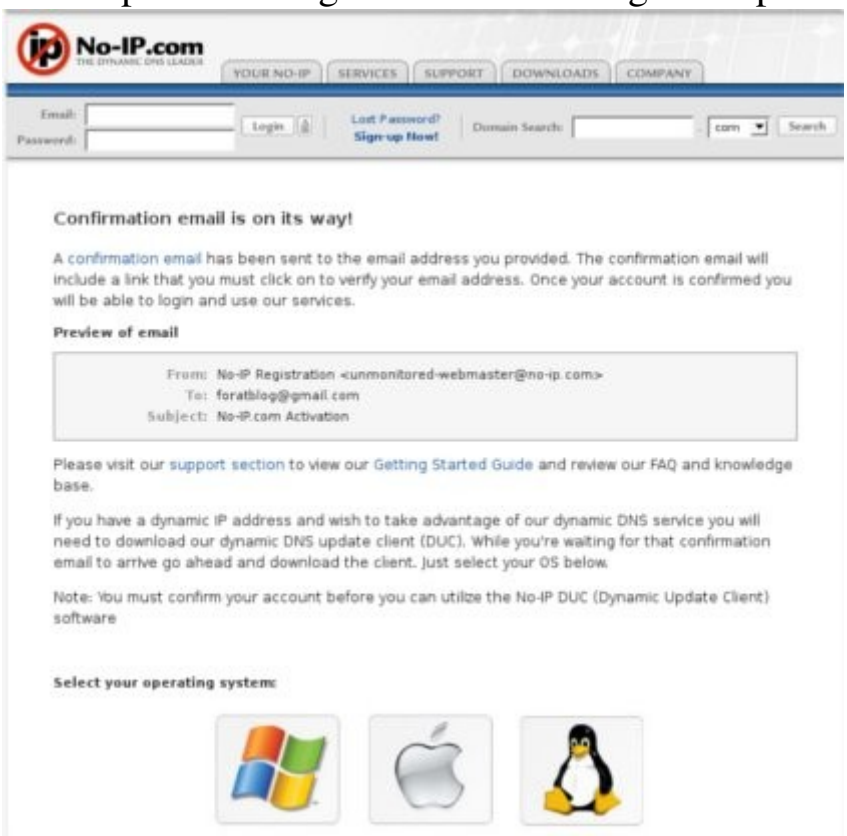
Una vez explicado que es y para que sirve el servicio de **No-IP** vamos a proceder con la instalación …

Abrimos un navegador e insertamos la dirección **http://www.no-ip.com/newUser.php** para registrarnos …



Debemos rellenar todas las casillas con datos verdaderos, no os preocupéis que no os van a vender nada si lo hacéis todo al pie de la letra de este manual.

Una vez completado el registro veremos la siguiente pantalla …



Aquí nos indican que nos han mandado una confirmación por email para verificar los datos con un enlace que tendremos que visitar obligatoriamente

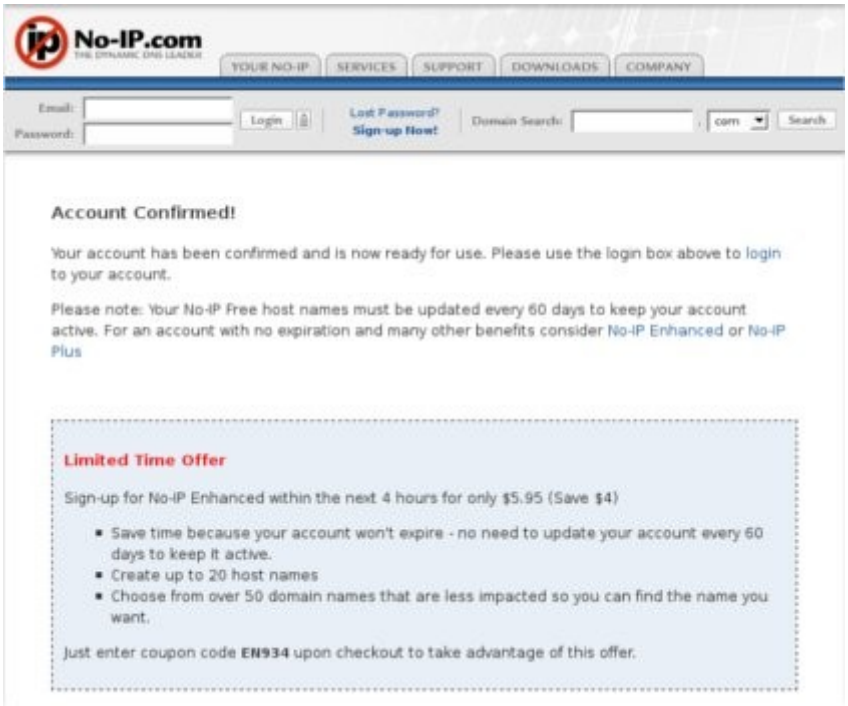
para activar nuestra cuenta.

Abrimos nuestro correo y buscamos la linea …

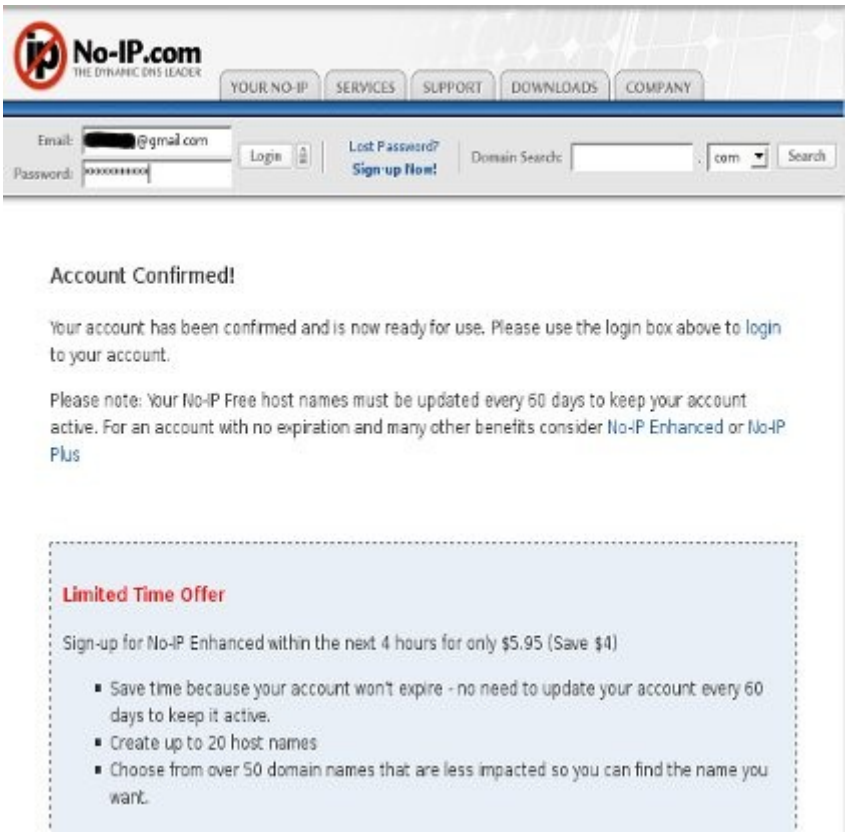
To activate your account please click the following URL:

y visitamos el siguiente enlace …

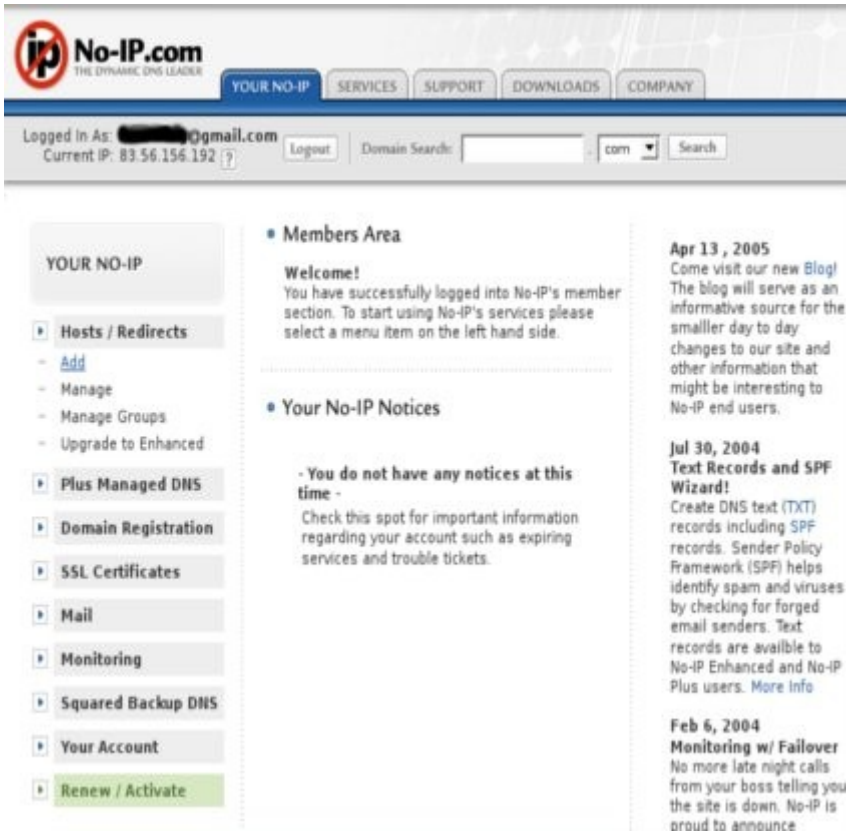
Una vez visitado el enlace de activación veremos algo así en el navegador …



Ahora rellenamos las casillas **nombre de usuario** y **password** con los datos que usamos para registrarnos …

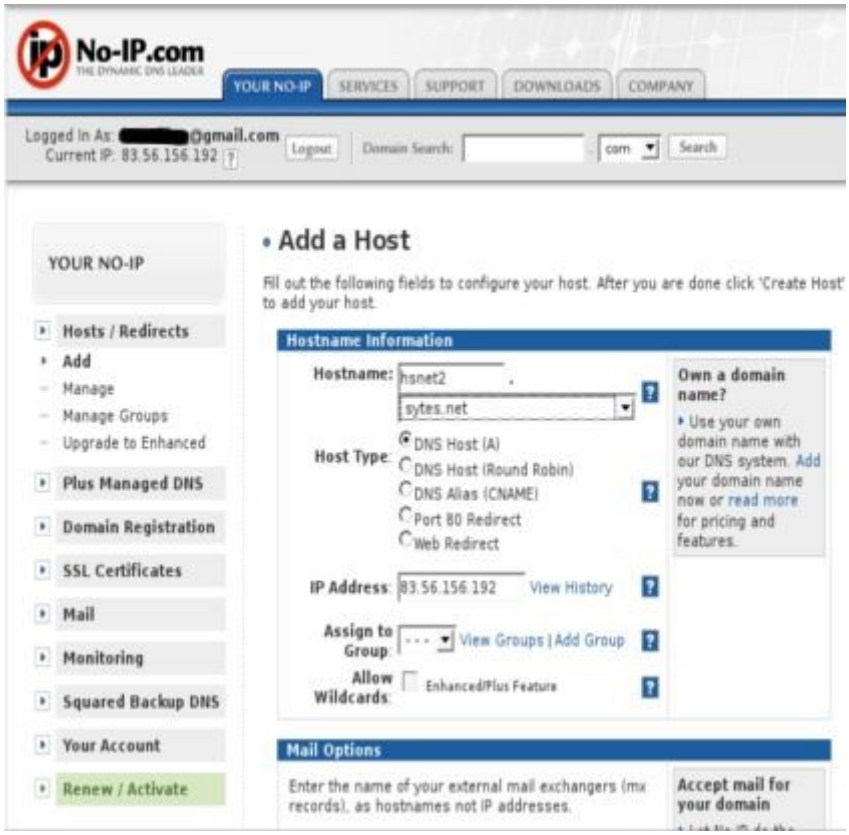


y pinchamos en (**Login**) …



Como podéis ver ya estamos dentro de el área de usuario y en la parte superior izquierda, debajo de nuestra dirección de correo esta la **IP** con la que salimos a Internet que es la misma con la que nuestro servidor puede ser localizado.

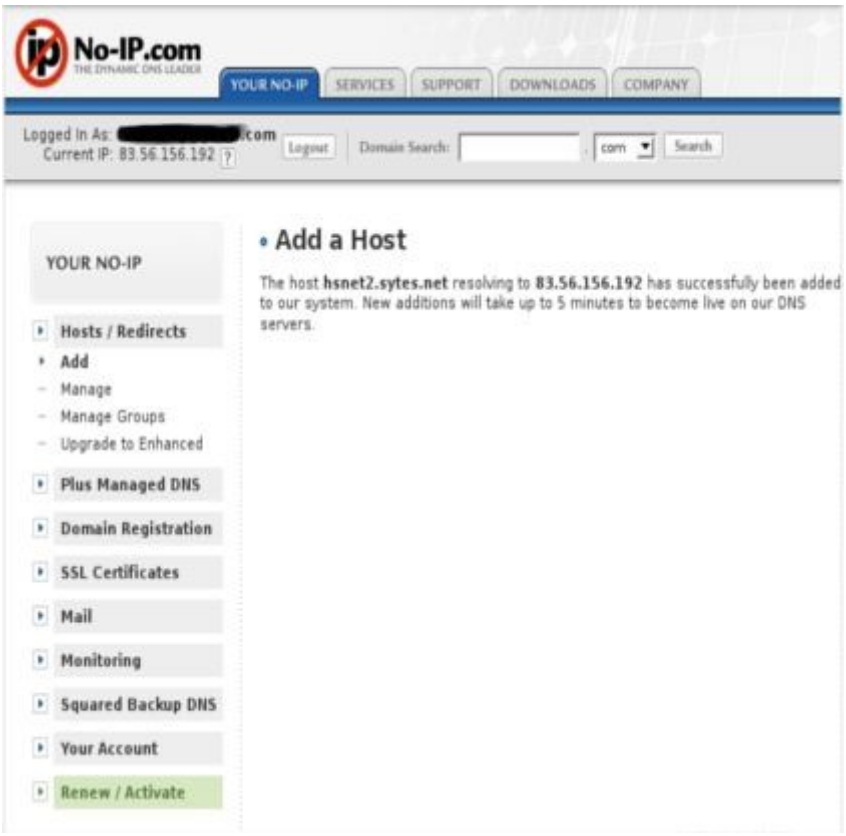
Vamos a registrar un dominio pinchando sobre el botón (**Add**) …



Rellenamos la casilla **Hostname** con el nombre de nuestro servidor y la casilla de abajo con el **subdominio** (**sytes.net**) Os recomiendo que elijáis este **subdominio** porque lo hemos introducido por varias partes de nuestro servidor y si elegimos otro podríamos tener problemas.

Como todos habéis podido ver a lo largo de este manual, mi servidor se llama **hsnet2** y mi dominio es **hsnet2.sytes.net**. *Siempre repito esto para que sirva de referencia.*

Una vez rellenadas estas dos casillas bajamos hasta el final de la pagina y pinchamos sobre el botón (**Create Host**), veremos una pantalla como esta …



Una vez llegamos a este punto podemos cerrar el navegador, arrancar nuestro **servidor en Linux Debian** y logearnos como (**root**).

Ahora instalaremos el software cliente No-IP para que mande los **pings** a nuestro **dominio.sytes.net** …

~ codigo ~

apt-get install no-ip

Una vez instalado vamos a configurarlo ejecutando …

~ codigo ~

no-ip -C

Y responderemos a unas cuantas preguntas … (*Las respuestas en* **negrita**)

- *Please enter the login/email string for no-ip.com* **Dirección de correo electronico** con la que nos registramos en No-ip.

- *Please enter the password for user 'foratblog@gmail.com'* **Password** con el que nos registramos en No-ip

- *Only one host [hsnet2.sytes.net] is registered to this account.*

It will be used.

- Please enter an update interval:[30] Pulsamos la tecla **Enter**

Do you wish to run something at successful update?[N] (y/N) Pulsamos la tecla **Enter**

New configuration file '/etc/no-ip.conf' created.

Ya tenemos configurado nuestro cliente **No-IP** que ira mandando **pings** cada 30 minutos con nuestra **IP** para que nuestro dominio (**dominio.sytes.net**) siempre sea relacionado con nuestra **IP** actual.

De momento no podréis ver el contenido de vuestro servidor web desde el exterior ya que todavía no hemos abierto los puertos en el **Router** para redireccionarlos hacia el servicio deseado.

Vamos a tener que abrir un par de puertos en nuestro **Router** para poder operar desde **Internet** con nuestro servidor.

Tenemos que redireccionar el puerto **80** hacia nuestro servidor para ofrecer contenido **Web** y el puerto 21 para poder acceder al **FTP**.

Esto es una cosa que os la explicaré en el siguiente articulo ya que creo que se merece un articulo para el solo.

Tened paciencia que ya prácticamente estamos listos para terminar.

Abriendo puertos en nuestro Router

En este articulo voy a explicaros para que sirve abrir puertos en un Router. Los servicios que nuestro servidor Web va a ofrecer hacia Internet son básicamente el Web y el Ftp.

Cada servicio que ofrecemos hacia Internet tiene asignado un puerto con el que se comunicará. Seguro que a mas de uno le suena aquello de … ¿ *Como abro los puertos del Router para el emule* ? Esta pregunta la he escuchado una infinidad de veces. Bueno lo que os intento explicar es que cuando al acceder a nuestro Router desde Internet con un navegador, el Router debe de tener el puerto 80 abierto y redireccionado hacia la IP de nuestro servidor y con el Ftp el 21.

Aquí os dejo un listado de puertos completo.

Cada Router tiene una interfaz gráfica diferente pero básicamente todos tienen las mismas características de configuración.

Para empezar tenemos que acceder a nuestro **Router** desde un navegador introduciendo su **IP local** que es lo mismo que nuestra puerta de enlace hacia **Internet** o **gateway.**

Automáticamente se nos pedirá el **nombre de usuario** y el **password** para acceder a el. Estos datos seguramente los encontrareis junto a la documentación del **Router**. Si no tenéis estos datos deberéis llamar a vuestro proveedor de Internet y pedírselos.

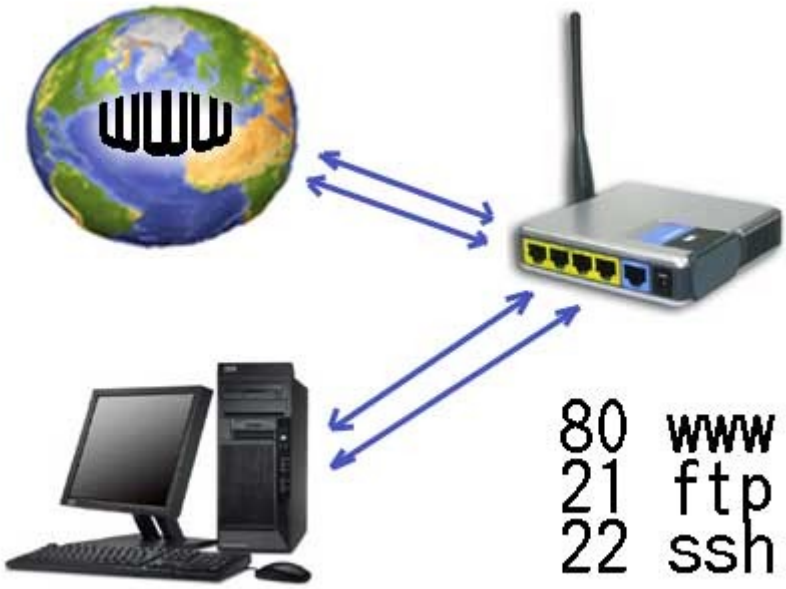
(*Tienes todo el derecho de poder acceder a un dispositivo que has comprado*)

Tengo entendido que **Imagenio** y algunos otros que se están apuntando al carro de tenernos controlados, tienen un portal especifico en **Internet** donde se pueden gestionar los puertos. Si no sabéis como, tendréis que llamar y preguntar como podéis acceder a el.

También os dejo aquí un enlace sobre los **passwords por defecto en diferentes Routers**, siempre cabe esta posibilidad.

Me es prácticamente imposible explicaros como hacer esta configuración ya que cada **Router** es diferente y yo no los tengo todos para probarlos y enseñaros como se hace.

Tan solo os puedo decir que tenéis que redireccionar el puerto **80** y el **21** hacia la Ip de vuestro servidor. (*en mi caso la 192.168.1.71*)



Necesitamos abrir estos dos puertos en nuestro **Router** obligatoriamente así que aplicad el cuento y googlear que seguro que alguien tiene un manual de el mismo **Router** que tenéis en casa.

Aquí os voy a dejar una pequeña recopilación de sitios que explican como abrir puertos en diferentes modelos de Router ADSL.

En **ADSLZONE** podemos encontrar manuales para todos estos Routers …

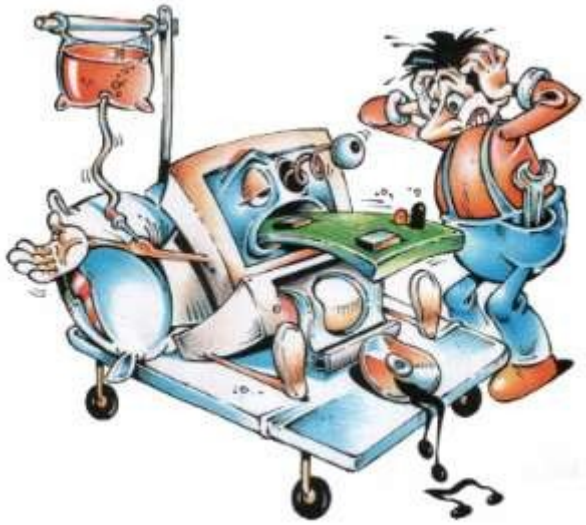
3Com 812, 3Com 11 G, 3Com, 3CRWDR100Y72, Alcatel Speed, Touch, Belkin 802.11B, Belkin 802.11G, Benq ESG 103, Cisco 827, CNet CAR-854, CNet CWR-854, Comtrend 500, Comtrend 535, Comtrend 536+, Comtrend 5071, Conceptronic C54APRA, Conceptronic CADSLR4, Conceptronic C54BRS4, Conexant CX82310, N&C RWGD45, D-Link 500, D-Link 504G, D-Link 504T, D-Link 524T, D-Link G604T, D-Link 614, D-Link 524, D-Link G624T, Huawei SmartAX, MT882, Huawei HG520, H. Echolife HG520V, Inventel DW-B-200, Linksys BEFSR41, Linksys WAG54G, Linksys WAG354G, Linksys WRT54G, Linksys AG241, Netgear DG834G, Nokia M1112, Ovislink DR-WG4S, SMC 7204 BRA, SMC 7804 WBRA, SMC 7401 BRA, Sagem Fast 1200, Sagem 1500, Speedstream 5200, Speedstream 5660, Supergrass DYN04+, Telsey CPVA500, Thomson 510, Thomson ST580i, Thomson 530, Thomson 545, Thomson 546, Thomson TcW 720, US Robotics 9003, US Robotics 9105, US Robotics 9110, DrayTek Vigor 2600, Webstar 2320, Xavi x7028r, Xavi x7768r, Xavi x7868r, Xavi x8121r, Zoom X5, Zoom X5 5554A, Zoom X6, Zyxel P623-41, Zyxel P643, Zyxel P645R-A1, Zyxel P650HW, Zyxel P660HW, Zyxel P660H-D1, Zyxel P660HW-D1, ZyAir G-2000

En **ADSLAYUDA** podemos encontrar manuales para estos Routers …

3Com 3CRWDR100Y72, 3Com Wireless 11g, Carpobox I, Conceptronic C54APRA, Comtrend 535, Comtrend 536+, D-Link G604T, Inventel DW-B-200, Inventel LiveBox, Linksys WAG354G, Linksys WAG54G, Linksys WRT54G, Linksys WRT54GX, Netgear DG834G, SMC 2084 WBRPG, SMC 7804 WBRA, SMC 7904 WBRA2, Sagem Fast 1500wg, Supergrass SGW ADSL2+, Thomson 580 i, US Robotics 8054, US Robotics 9106, Xavi 7768, Zyxel 650/660HW, 3Com 812, Alcatel, Speed Touch, Cisco 827, Comtrend 500, Conectabox 1000 B, D-Link 500, D-Link 504G, D-Link 504T, Huawei MT882r-T, Linksys BEFSR41, Nokia M1112, Sagem Fast 1200, SpeedStream 5660, Supergrass DYN04PLUS, Telsey CPVA 500, Thomson 510, US Robotics 9003, US Robotics 9105, Xavi 7028, Zoom x5, Zyxel 643

Si no os aclaráis de como hacerlo podéis poner en los comentarios vuestro modelo de **Router** y nombre de vuestro proveedor de Internet e intentaré ayudaros. Suerte

Últimos ajustes



Uno de los artículos finales de este gran manual será el como montar un **Blog** con **Wordpress** pero antes de esto vamos a ajustar un poco la configuración del servidor mínimamente para que todo quede mas pulido además de ordenado.

Con el siguiente manual crearemos un usuario especifico que sera el encargado de el directorio raíz en nuestro servidor Web. También eliminaremos la extensión **www** pudiéndose ver desde el exterior nuestro **dominio.sytes.net** en vez de **dominio.sytes.net/www/**.

También modificaremos el archivo **hosts** del sistema para dar un aspecto mas limpio a nuestra maquina en programas como el **phpsysinfo** y en **Awstats** que nos muestra por defecto **maquina.dominio.sytes.net** y vamos a dejarlo como **dominio.sytes.net**.

Para esto vamos a ajustar un poco la configuración del sistema arrancando nuestro **servidor en Linux Debian** y logeandonos como (**root**). Comenzaremos por crear un usuario para el contenido web de nuestro servidor

” *Así no mezclaremos cosas* “.

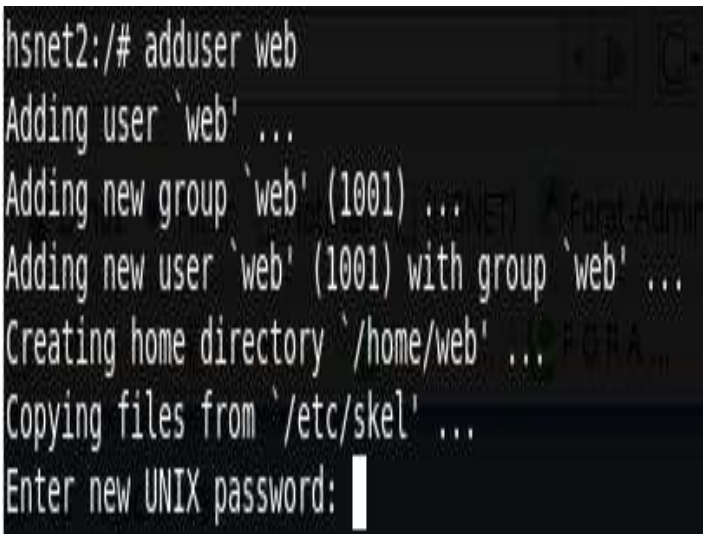
En mi caso voy a crear un usuario llamado (**web**) y su directorio raíz dentro de **/home/web/** será donde hospedare el contenido web que quiero que los usuarios vean. Con este usuario también tendremos acceso al **FTP** para poder subir y bajar archivos directamente.

Creamos el usuario …

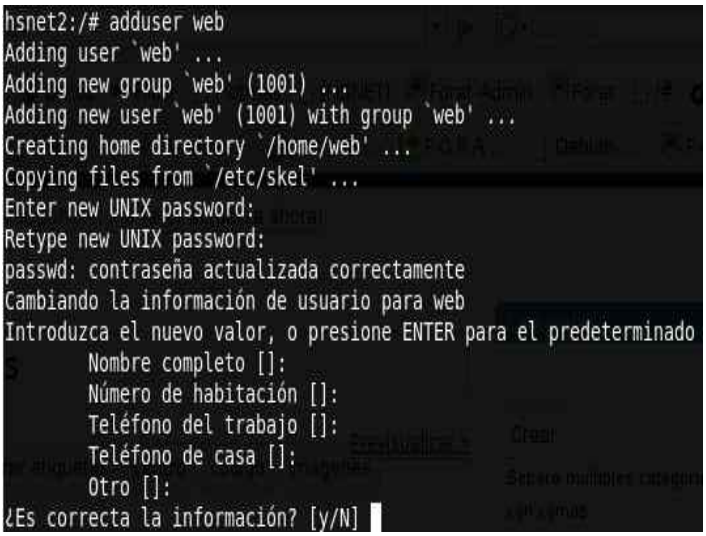
- codigo -

adduser nombredeusuario

El comando **adduser** creará automaticamente los directorios necesarios para el usuario incluyendo **/home/usuario** …



Seguidamente introducimos el **password** para el usuario y lo volveremos a introducir para verificar de que no nos hemos equivocado.
Una vez introducido nos pregunta algunos datos sobre el usuario, esto es opcional, tan solo pulsa la tecla (**Enter**) y lo dejamos en blanco si no queremos poner nada …



Introducimos (**Y**) y listo. Ahora tenemos un usuario creado que vamos a destinar para la administración del apartado web en este servidor.

Ahora vamos a retocar el **servidor Web Apache2** para que el directorio **/home/usuarioweb** sea el predeterminado y podamos operar directamente desde el.

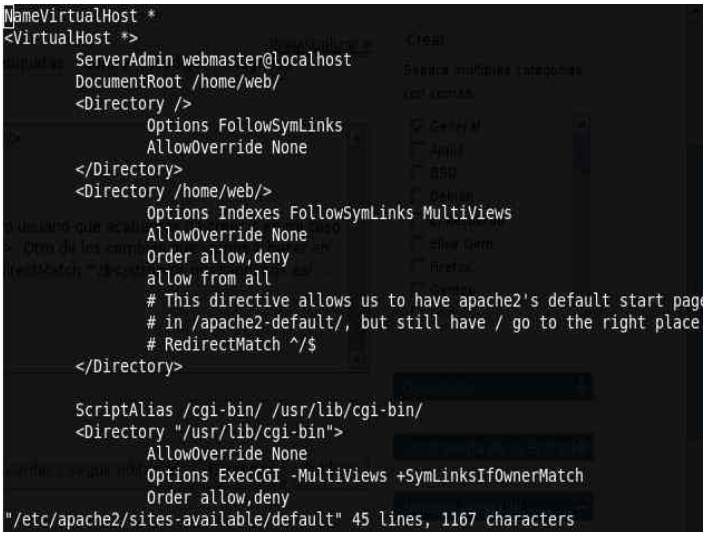
Esta configuración eliminara la extensión (**www**) que ahora mismo tenéis configurada en vuestros servidores pudiendo ver el contenido Web del servidor directamente desde vuestra **IP local** o **dominio**, (En mi caso **http://192.168.1.71** o **hsnet2.sytes.net**)

Editamos el fichero (**default**) dentro de la ruta **/etc/apache2/sites-available/** con el editor **vi** …

- codigo -

vi /etc/apache2/sites-available/default

Una vez dentro cambiaremos la ruta **/home/usuario/** por la de el nuevo usuarioweb que acabamos de crear, (en mi caso **/home/usuarioweb/**) en la linea **DocumentRoot** y **Directory**. Otro de los cambios que vamos a hacer en este fichero es el de comentar la linea **RedirectMatch ^/$** con un símbolo (**#**) quedándonos así …



Una vez editado, guardamos y salimos del editor con (**ESC**) y (**:wq!**).

Ahora vamos a crear los enlaces simbólicos hacia la ruta **/home/usuarioweb** de los programas **phpsysinfo**, **phpmyadmin** y **cgi-bin** con …

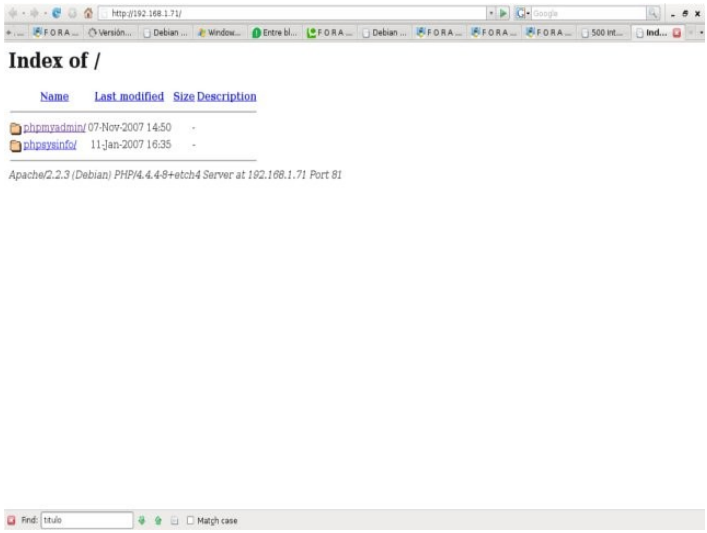
- codigo -

ln -s /var/www/phpsysinfo /home/usuarioweb/phpsysinfo

ln -s /var/www/phpmyadmin /home/usarioweb/phpmyadmin

ln -s /usr/lib/cgi-bin /home/usuarioweb/cgi-bin

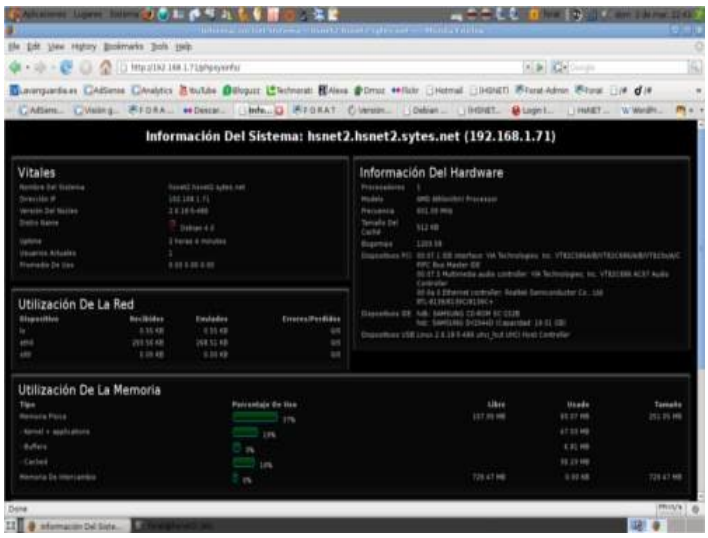
Ahora para ver si todo ha ido bien, vamos a abrir un navegador desde otro ordenador y vamos a introducir solo la **IP local** del nuestro servidor Web …



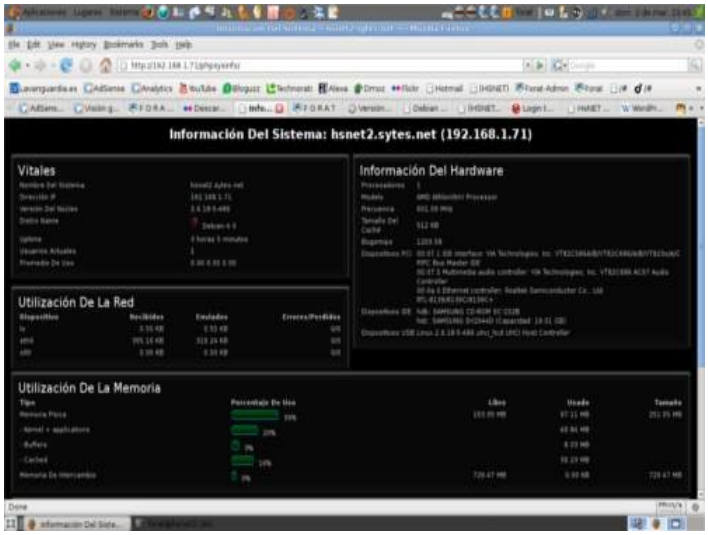
Como podéis comprobar ahora ya no veremos el directorio (**www**) junto a la **IP local**, ahora cuando montemos la Web o Blog principal del servidor, solo tendremos que hospedarlo en el directorio raíz **/home/usuarioweb**.

Por otro lado también vamos a configurar el fichero **hosts** para que muestre solo el nombre de dominio y no también el de nuestra maquina. Esto no es algo especialmente importante pero queda mucho mejor a la vista de algún programa como el **phpsysinfo**.

Si nos fijamos vemos el nombre **hsnet2** sale repetido dos veces delante de **sytes.net**. Esto no es un fallo, tan solo es que hemos configurado al inicio el nombre de nuestro servidor y el nombre de dominio con el mismo nombre. No es un problema, es mas, es completamente necesario cuando tenemos mas de un servidor en casa, pero para los que tenéis solo un servidor puede quedaros un poco mas limpio y pulido, fijaos así es como lo tenemos ahora …



y así es como nos va a quedar después de una pequeña modificación …



Para hacer esto editamos el archivo **hosts** dentro de la ruta **/etc/** y modificamos ….

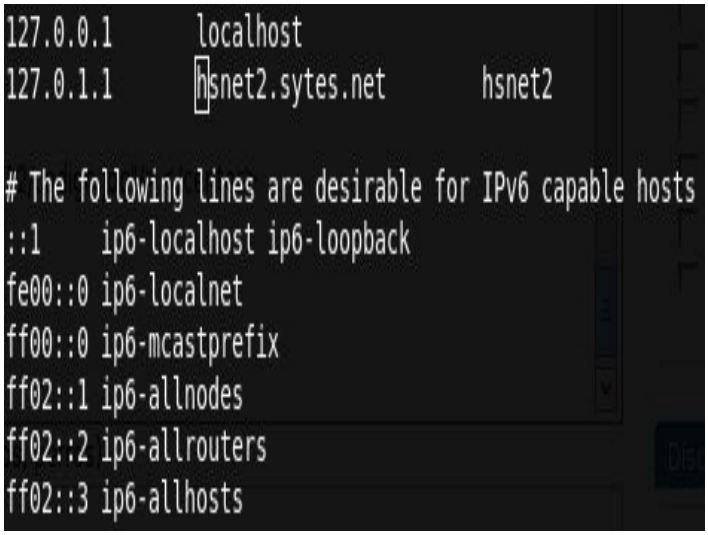
- codigo -

vi /etc/hosts

```
127.0.0.1       localhost
127.0.1.1       hsnet2.hsnet2.sytes.net hsnet2

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
```

Borramos el nombre de vuestra maquina que sale repetido quedando así …



Guardamos y salimos con (**ESC**) y (**:wq!**).

Se que muchos estáis esperando el articulo para crear vuestro propio **Blog** dentro de este servidor que nos estamos montando, así que doy por finalizado este articulo.

Montando un Blog con Wordpress

Montar un Blog va a ser el ultimo paso que vamos a dar antes de acabar este **servidor Web en Linux Debian** que nos estamos montando. Para ello vamos a usar el software de **Wordpress** que es con el que he montado yo mismo este mi Blog **Forat.Info**.

Para comenzar vamos a descargarnos **Wordpress** y lo vamos a descomprimir en el directorio raíz de nuestro servidor Web para que pueda ser visto por los usuarios que accedan a el desde el exterior.

Para ello arrancamos nuestro **servidor Web con Linux Debian**, nos logeamos como (**root**) y vamos a entrar en el directorio raíz de nuestro servidor Web (en mi caso **/home/web/**), después procedemos a su descarga con el comando (**wget**) de la siguiente forma …

~ codigo ~

cd /home/web

wget http://wordpress.org/latest.tar.gz

Ahora lo descomprimimos con …

~ codigo ~

tar xvzf latest.tar.gz

Una vez descomprimido nos creara una carpeta llamada (**wordpress**), yo personalmente voy a copiar todo el contenido de dicha carpeta en el directorio raíz para que al cargar la IP de mi servidor (**http://192.168.1.71**) o mi dominio

(**http://hsnet2.sytes.net**) podamos ver el Blog en vez de el contenido en directorios.

Pensad que cada vez que descomprimamos el archivo **latest.tar.gz** nos creara una carpeta llamada **wordpress**, así que si quieres tener mas de un **Blog** deberíais renombrarla por el nombre que queramos para el nuevo **Blog**.

Yo voy a montar un **Blog** en el directorio raíz del servidor Web y también me voy a montar otro llamado (**Dummy**) en una carpeta.

Para empezar copiamos todo el contenido de la carpeta (**wordpress**) en el directorio raíz con …

- codigo -

cd wordpress
cp -r . /home/web/

Y ahora renombramos la carpeta (**wordpress**) para crearnos un segundo Blog (ponerle el nombre que queráis, yo le pondre Dummy) …

- codigo -

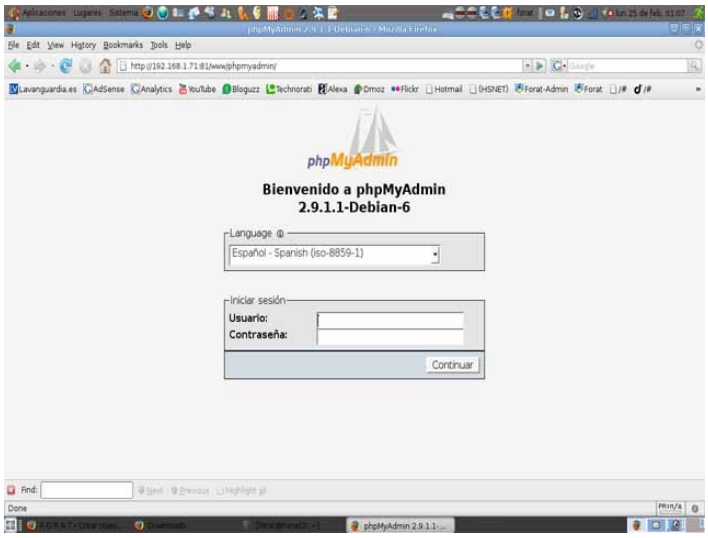
mv wordpress dummy

Seguidamente vamos a crear la base de datos con **PhpMyAdmin**. Abrimos un navegador e introducimos …

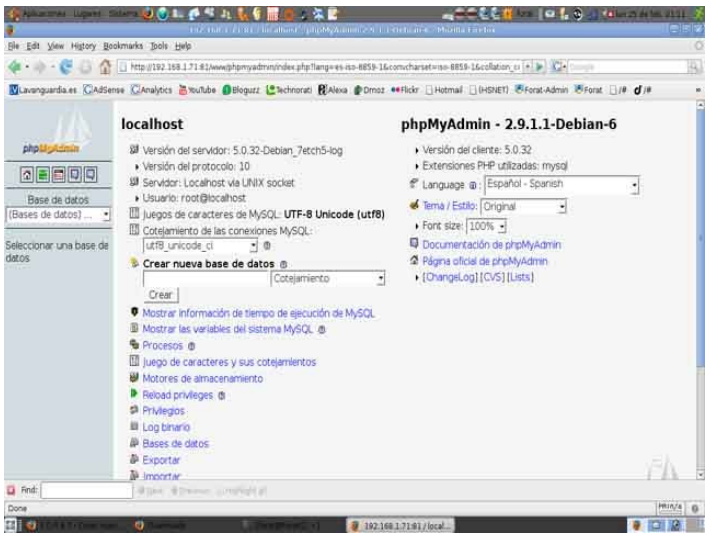
- codigo -

http://ipdetuserver/phmyadmin/

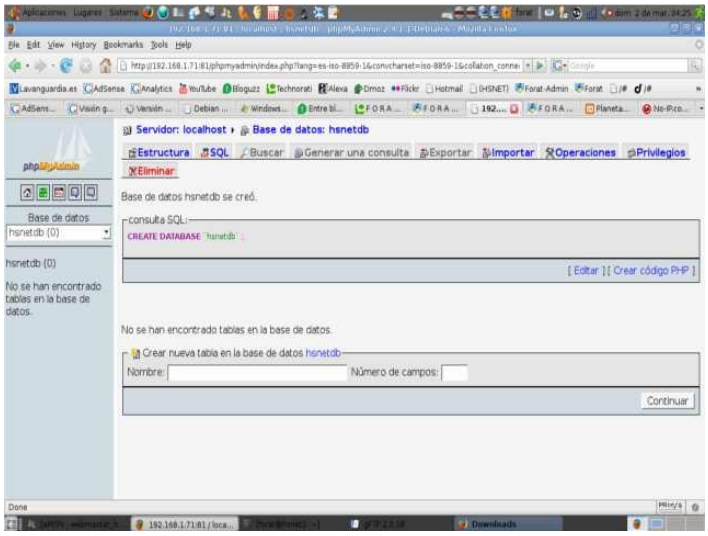
Veremos algo así …



Introducimos el **nombre de usuario** y **password** de **root** que asignamos anteriormente en el articulo (**Bases de datos MySQL**) y creamos la base de datos rellenando la casilla (**Crear nueva base de datos**) …



Una vez creada veremos este mensaje de confirmación …



Una vez creada volvemos al servidor Web y vamos a renombrar el archivo **wp-config-sample.php** y lo vamos a llamar **wp-config.php** en nuestro directorio raíz donde guardamos el **wordpress**. (en mi caso **/home/web/**) …

~ codigo ~

cd /home/web

cp wp-config-sample.php wp-config.php

Ahora lo editamos para asignarle la base que antes creamos con **phpmyadmin** …

~ codigo ~

vi wp-config.php

Modificamos estas tres lineas asignándole la **base de datos**, **nombre de usuario** y **password** …

~ codigo ~

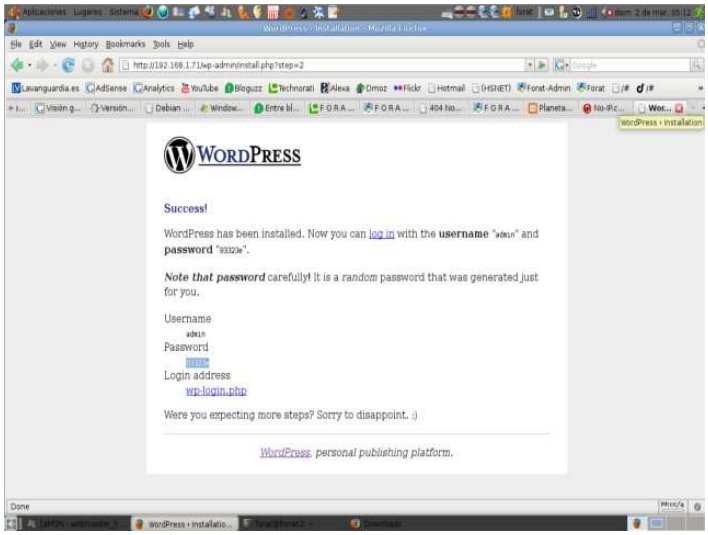
define('DB_NAME', 'nombredelabasededatos'); // The name of the

database
define('DB_USER', 'nombreeusuaio'); // Your MySQL username
define('DB_PASSWORD', 'passworddeusuario'); // …and password

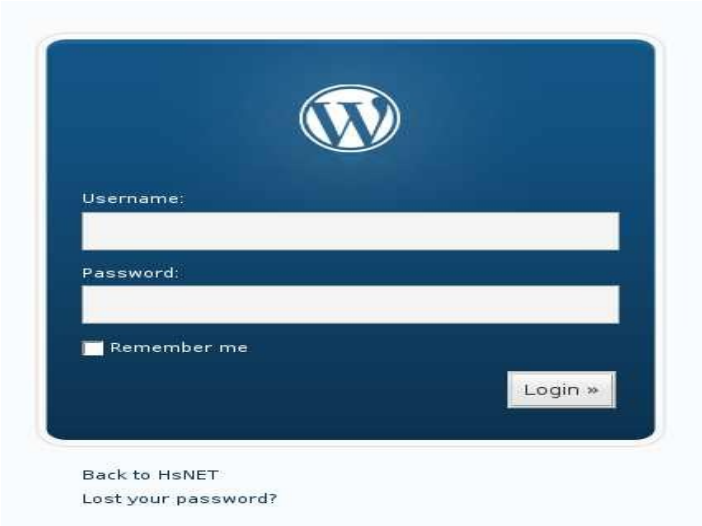
Una vez introducidos los datos correctamente guardamos y salimos del editor (**vi**) con la tecla (**ESC**) y (**:wq!**)

Ahora vamos a otro ordenador que tengamos en nuestra red e introducimos en el navegador la dirección de tu servidor adjunto del fichero **install.php** que nos servirá para instalar el **Wordpress** (en mi caso **http://192.168.1.71/wp-admin/install.php**) …

Introducimos el titulo del **Blog**, el correo electrónico y pinchamos sobre **install wordpress** …



Como podemos ver el programa nos asigna un **password** para el usuario **admin**. Copia el **password** y pincha sobre **wp-login.php** …



Ahora debemos introducir como nombre de usuario (**admin**) y pegamos el **password** que antes copiamos en la casilla **password** …

y ya estamos dentro del panel de administración

Ahora al introducir **http://dominio.sytes.net** veremos este **Blog** que acabamos de configurar en la raíz del servidor Web.

Recuerda cambiar el **password** de el usuario (**admin**) dentro del panel de **Wordpress** accediendo a la sección de usuarios.

Para configurar otros **Blogs** dentro de nuestro servidor Web tan solo tendremos que repetir estos mismos pasos y listo. Podemos montarnos todos los Blogs que queramos, simplemente para practicar, editar themes o enseñar nuestro Blog a nuestros visitantes.