



TFC: Plataforma GNU/Linux

Proyecto: Debian con LTSP integrado

Consultor: Miquel Àngel Senar Rosell

Índice de contenidos

1 Licencia.....	3
2 Resumen del documento.....	4
3 Descripción general del proyecto.....	5
4 Introducción	6
4.1 Definición del proyecto.....	7
4.2 Motivaciones.....	8
4.3 Viabilidad, alcance y objetivos.....	9
4.4 Requisitos.....	10
4.5 Planificación temporal.....	12
5 Diseño del sistema.....	13
5.1 Fase 0: Requisitos.....	13
5.2 Fase 1: Instalación.....	14
5.3 Fase 2: Configuración.....	17
6. Finalización.....	20
6.1 Creación del CD a partir del diseño.....	20
7 Pruebas.....	27
8 Errores encontrados y soluciones.....	28
8.1 Al crear el CD con simple-cdd fallan paquetes al descargarse.....	28
9 Mejoras y conclusiones.....	30
10 Anexos.....	33
10.1 Proceso de arranque de un cliente con LTSP.....	33
10.2 Creación masiva de usuarios.....	34
11 Bibliografía.....	35

Miguel Ángel Madrid Salinas

1 Licencia

Creative Commons

Este documento está bajo la licencia Creative Commons como Reconocimiento – NoComercial – Compartirlgual. No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.
<http://creativecommons.org/licenses/by-nc-sa/2.5/es/>

2 Resumen del documento

Este documento presentará un proyecto para la realización de un DVD personalizado enfocado a la educación. Estará basado en un sistema operativo Debian y en el proyecto LTSP (Linux Terminal Server Project). A lo largo del mismo se irán desarrollando una serie de apartados explicando cada uno de los puntos.

Una introducción conceptual del proyecto y de todas las herramientas utilizadas ayudará a evaluar las ventajas y desventajas de llevarlo a cabo. Más adelante, se diseñará el servidor LTSP con todos los paquetes educativos necesarios para su posterior configuración. Por último, se creará el DVD personalizado basándose en este diseño y se realizarán las pruebas necesarias para obtener unos resultados y unas conclusiones finales.

3 Descripción general del proyecto

Este proyecto se basará en la realización de un DVD Debian personalizado con *LTSP* integrado. Para llevarlo a cabo, antes se instalará un sistema operativo Debian en un entorno de pruebas. Además, se instalará un servidor de terminales Linux (LTSP) y una serie de paquetes educativos. Esta lista de paquetes y la configuración del servidor servirá para crear un DVD personalizado disponible para una futura instalación en un servidor de producción.

Miguel Ángel Madrid Salinas

4 Introducción

4.1 Descripción del proyecto

4.2 Motivaciones

4.3 Viabilidad, alcance y objetivos

4.4 Requisitos

4.5 Planificación temporal

4.1 Definición del proyecto

El proyecto estará diseñado sobre una instalación del sistema operativo GNU/Linux Debian, en un entorno de pruebas.

El sistema operativo *Debian GNU/Linux* es una distribución del sistema operativo Linux, creado por Ian Murdock en 1993, apoyado por el proyecto *GNU* de la *FSF* (Free Software Foundation). Es también un conjunto de programas que se ejecutan sobre él (GNU y otros). A día de hoy existen casi 30000 paquetes en sus repositorios. Además, es libre de usar y distribuir, no hace falta pagar por usarlo. Todos los paquetes que incluye por defecto cumplen la licencia GNU GPL aunque existen repositorios para los paquetes no oficiales, dentro de las secciones non-free y contrib. Por otro lado, su desarrollo es continuo, hay cerca de 1000 voluntarios trabajando en él cada día, por lo que su mejora es constante y rápida. Las actualizaciones son continuas, e incluso algunas suceden cada día.

Una vez instalado el sistema operativo Debian se procederá a la instalación de LTSP, un servidor de terminales de código abierto que permite conectar a él de forma simultánea un número determinado de equipos clientes, también llamados *thin clients*. Estos equipos ejecutan las aplicaciones desde el servidor al cual se conectan normalmente por *ethernet*.

En el momento de este documento, la última versión de *LTSP* disponible es la v5. Ésta está formada por varios servicios o *daemons*, entre los que destacan:

- *DHCPd: Dynamic Host Configuration Protocol daemon*. El servidor que permite a los clientes de una red obtener una IP de forma dinámica.
- *TFTPd: Trivial File Transfer Protocol daemon*. El servidor para la transferencia de archivos, ideal para transferir archivos de tamaño reducido.
- *NFSd: Network File System daemon*. El servidor de sistema de archivos de red permitirá a los clientes acceder a ficheros remotos en el servidor como si fueran locales.
- *SSHD: Secure SHell daemon*. El intérprete de órdenes segura, en LTSP se usa para cifrar la sesión del cliente en el servidor.

La tecnología Terminal Server ya se usaba en otras plataformas:

- En la parte cliente se usan los dispositivos de entrada-salida como la pantalla, el teclado o el ratón.
- En la parte servidor se guardan todos los datos, se instalan las aplicaciones, se ejecutan en él y se muestran en la pantalla del cliente.

Por esta razón, el cliente puede ser un equipo con componentes básicos o reciclados de otros. Todo el trabajo procesal se lleva a cabo en el servidor. No necesitarán disco duro, tan solo habrá que tener en cuenta la integridad de los datos en el servidor, si un equipo falla es fácilmente reemplazable sin necesidad de restaurar sus datos.

Una vez instalado y configurado, tanto Debian como LTSP, se instalarán una serie de paquetes educativos. Este listado servirá más tarde para la creación del DVD personalizado.

4.2 Motivaciones

Existen distribuciones como Skolelinux (Debian) y Edubuntu (Ubuntu) que ya vienen preparadas con LTSP. Una de las razones por las que me he decantado por este proyecto es el poder crear una distribución orientada a la educación desde cero, estable, compacta y totalmente libre. Otras distribuciones, como las antes mencionadas, cargan paquetes que no siempre se van a utilizar.

Por su valor educativo. Un entorno LTSP, junto con la naturaleza libre de GNU/Linux, ayuda al aprendizaje ya que permite que las aplicaciones que se usan en la escuela, academia o lugar de trabajo puedan usarse fuera de éstos ámbitos, instalando o por medio de un LiveCD, cualquier distribución GNU/Linux compatible con esas aplicaciones.

Por su valor ecológico y de bajo coste. Este sistema es ecológico, por medio del reciclaje de equipos obsoletos se evita contribuir al deshecho electrónico. Estos equipos consumen menos energía que algunos equipos actuales por lo que el ahorro energético es también mayor. Así mismo, este sistema permite un ahorro económico. No es necesario pagar por licencias de uso, ni el sistema operativo GNU/Linux, eligiendo una distribución libre, ni por el uso del entorno LTSP.

4.3 Viabilidad, alcance y objetivos

Hay varios aspectos a tener en cuenta a la hora de elegir esta tecnología, entre sus ventajas destaca la orientación que tiene, está implementada para su uso en escuelas, aulas de formación, etc. Seguro, sencillo de administrar, respetuoso con el medio ambiente, altamente soportado por la comunidad de usuarios y/o desarrolladores; y es de bajo coste.

Este proyecto es rentable y de bajo coste. Aquí lo que se vende no es el producto, sino el posible servicio. El producto está orientado a aulas de formación, escuelas, ONGs, etc. Estos lugares disponen en muchas ocasiones de la infraestructura necesaria para implantar este proyecto, se les ofrece el DVD de instalación y la pueden realizar ellos mismos. Si no fuera así existen alternativas:

- Contratar el servicio de ADSL por medio de un ISP.
- Solicitar equipos donados por administraciones públicas, empresas, etc. Para usarlos como thin clients. Si tuvieran que comprarlo el coste orientativo sería:

Caja AT/ATX :	20€
Pantalla CRT:	15€
Teclado/Ratón:	10€
RAM:	15€
Sonido:	10€
Red:	10€
Cableado:	20€
Total:	100€ (20 equipos: 2000€)

Existiría la posibilidad también de recomendarles las características del servidor, pudiéndoles ofrecer un servidor preinstalado con *LTSP* a un coste final o mensual, así como el servicio técnico. Unos 1000€ sería el coste real del equipo servidor.

4.4 Requisitos

El despliegue de este proyecto en un entorno global implicaría tener en cuenta una serie de requisitos, en cuanto al *hardware*. Por un lado, la parte de los clientes, el servidor y también la infraestructura de red. Toda esta información está explicada y recomendada en la documentación de *LTSP*. A continuación explicaré estas recomendaciones.

4.4.1 Requisitos del cliente

El objetivo principal es poder reutilizar el material informático que dispone el departamento en el que se vaya a implantar este sistema. Aunque habrá que tener en cuenta una serie de requisitos mínimos que han de cumplir estos equipos, a parte del teclado, ratón y pantalla; es necesario saber qué CPU, memoria RAM, tarjeta de red y tarjeta de video es la idónea en cada escenario.

CPU

Es necesario al menos una CPU a 533MHz, para el modo seguro de acceso a LDM (Administrador de pantalla de LTSP), ésto es necesario cuando se usa la comunicación cifrada por medio de SSH. Si se desactiva esta opción, la CPU puede ser más lenta, hasta 233MHz.

Memoria RAM

El mínimo para que el cliente pueda arrancar es de 48MB, aunque es recomendable 128MB de RAM. Con 512MB nos aseguramos un arranque más rápido.

Tarjeta de red

El cliente necesita cargar el sistema por red, para ello usa un programa interno en la tarjeta llamado "cargador de arranque por red" (*network boot loader*). Otras tarjetas más antiguas no disponen de este programa interno y será necesario cargarlo por floppy, cd-rom o usb.

Los cargadores de arranque por red más utilizados son:

PXE:

Es el más común, la mayoría de las tarjetas y placas base con red integradas lo soportan. Si la tarjeta dispone de PXE, será capaz de arrancar sin problemas. La ROM de arranque PXE puede cargar de una vez un archivo de 32KB, el kernel de linux es más grande que eso, para ello carga un archivo llamado pxelinux, que contiene las instrucciones para poder cargar el kernel de linux.

Etherboot:

Es el equivalente de software libre, sirve para aquellas tarjetas que no dispongan de PXE. Puede arrancarse desde floppy, usb o cd-rom o, incluso, grabado en la EPROM de la tarjeta si ésta dispone de un zócalo para ello.

Yaboot: Su nombre viene del acrónimo "Yet Another Boot Loader" y es especial para las máquinas PowerPC de Macintosh.

Tarjeta de video

Cualquier tarjeta de video de bus PCI con 16MB o más de memoria será suficiente.

4.4.2 Requisitos del servidor

Una vez aclaradas las recomendaciones para los clientes es importante decidir qué requisitos mínimos ha de cumplir el servidor en cuanto a *CPU*, memoria *RAM* o discos duros. Estos valores están ya calculados en la documentación de *LTSP*, a modo orientativo.

CPU

La velocidad del procesador dependerá de que programas se vayan a usar y el número de clientes conectados a la vez. Es preferible a partir de 3GHz. Con 30 clientes o más es recomendable un procesador de 4 núcleos, como los Intel core i5 o i7. También, si el número de clientes es elevado es recomendable segmentar la red utilizando un segundo servidor para conectar a él algunos equipos.

Memoria RAM

La memoria *RAM* del servidor es compartida por los usuarios conectados a él por medio de los clientes. La fórmula más usada para calcular esta cantidad es la siguiente:

$$256 + (192 * \text{usuarios}) \text{ MB}$$

Indica que habrá 256MB para el servidor y lo demás disponible para los clientes. Es decir, si queremos que al menos haya 20 clientes conectados al servidor necesitaremos un mínimo de:

$$256 + (192 * 20) = 256 + 3840 = \mathbf{4096 \text{ MB de RAM}}$$

Por tanto, teniendo esta cantidad de memoria podemos asegurarnos de que los clientes harán uso de ella y no de la memoria *swap* del servidor, ésta alojada en su disco duro, mucho más lenta que la memoria *RAM*, con la bajada de rendimiento que ésto implicaría.

A pesar de estas recomendaciones, si el uso del sistema es para un uso gráfico considerable es preferible doblar esta cantidad.

Discos duros

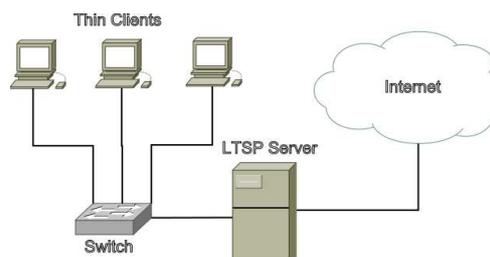
Es recomendable el uso de algún tipo de *RAID* en el servidor. Con *RAID1* y dos discos se salvaguardan los datos si uno de ellos falla. Tan sólo sería necesario sustituirlo por otro de las mismas características.

Por tanto, la configuración mínima, en un entorno real, para un buen rendimiento, serían dos discos SATA de 7200-10000RPM en *RAID1*.

4.4.3 Requisitos de red

Las recomendaciones mínimas de la infraestructura de red estará formada por un *Switch Gigabit*. Es preferible usar un *Switch* en lugar de un *Hub*, permitirá un mejor rendimiento. Se conectarían a éste unos 20 clientes y al servidor por medio de una de las tarjetas Gigabit. Una segunda tarjeta de red daría servicio de internet.

Por tanto, el entorno *LTSP* quedaría reflejado de la siguiente manera:



4.5 Planificación temporal

Tarea	Fecha Inicio	Fecha Fin
Introducción, requisitos, motivaciones, viabilidad	21/03/2012	25/03/2012
Diseño	26/03/2012	01/05/2012
Requisitos	26/03/2012	26/03/2012
Descarga de Debian Squeeze	26/03/2012	26/03/2012
Configuración de máquina virtual	26/03/2012	26/03/2012
Instalación	27/03/2012	
Instalación de sistema operativo Debian Squeeze en máquina virtual	27/03/2012	28/03/2012
Instalación de LTSP	29/03/2012	09/04/2012
Instalación de paquetes educativos	10/04/2012	14/04/2012
Configuración	15/04/2012	
Configuración de LTSP y todos sus servicios	15/04/2012	21/04/2012
Configurar sistema	22/04/2012	01/05/2012
Implementación	02/05/2012	27/05/2012
Crear CD de instalación Debian con LTSP integrado	02/05/2012	
Generar archivo Preseed	02/05/2012	09/05/2012
Generar script para postinstalación	10/05/2012	11/05/2012
simple-cdd	14/05/2012	27/05/2012
Pruebas	28/05/2012	03/06/2012
Conclusiones	04/06/2012	08/06/2012

5 Diseño del sistema

Una vez que se ha evaluado el proyecto se procede al diseño del sistema. Estará formado por los requisitos, la descarga de la imagen de Debian y la configuración de una máquina virtual; la instalación del sistema operativo, del servidor LTSP y de los paquetes educativos y de sistema; por último, se configurará

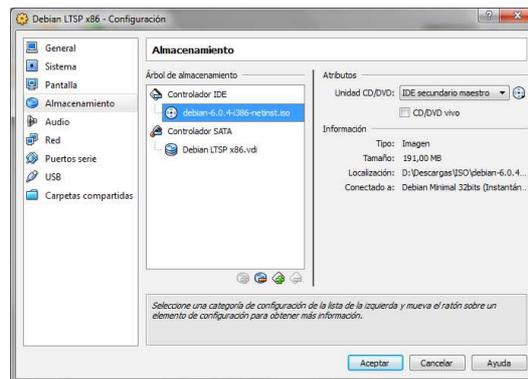
5.1 Fase 0: Requisitos

Hay varias versiones de Debian, he optado por la versión estable instalable por red, para plataforma x86, ya que ésta es compatible con el mayor número de aplicaciones:

<http://www.debian.org/distrib/netinst>

Una vez que está descargada la ejecuto desde una máquina virtual con el programa *Oracle VirtualBox*.

<https://www.virtualbox.org/wiki/Downloads>



Esta máquina virtual tendrá un máximo de 40GB de disco duro, 1GB de memoria RAM, VGA de 16MB y dos tarjetas de red Intel. Por otro lado, también se configura una máquina virtual para el cliente. Dispondrá de una tarjeta de red y ésta será prioritaria en el orden de arranque.

5.2 Fase 1: Instalación

Se procede a la instalación del sistema operativo Debian.



La instalación guiará por una serie de pasos en los que habrá que seleccionar:

- El idioma ("es_ES")
- La distribución del teclado ("es_ES")
- La tarjeta de red primaria para instalación (eth0)
- Nombre del equipo ("debian")
- Nombre del dominio ("ltsp.org")
- Nuevo usuario ("testltsp")
- Ubicación de zona horaria ("Península")
- Particionado del disco
- Réplica de Debian ("ftp.es.debian.org")
- Programas a instalar ("Utilidades estándar de sistema" y "Entorno gráfico")
- Instalación de paquetes

Una vez finalizada la instalación del sistema operativo, se inicia la instalación del servidor LTSP, por medio del siguiente comando:

```
# apt-get install ltsp-server-standalone
```

Este paquete instalará todos los servicios necesarios por medio de sus dependencias. Destacan las siguientes:

```
- dhcp3-server
  Servidor de DHCP para la asignación automática de direcciones IP

- libasound2-plugins
  Librería adicional de plugins para ALSA

- ltsp-server
  Entorno básico del servidor LTSP

- ltspfs
  Fuse based remote filesystem for LTSP thin clients

- nbd-server (>= 1:2.9.6-2)
  the Network Block Device server

- nfs-kernel-server
  Infraestructura del servidor de NFS del núcleo

- openbsd-inetd
  El superservidor de internet de OpenBSD

- openssh-server
```

Miguel Ángel Madrid Salinas

servidor de «secure shell», un reemplazo de rshd

- *squashfs-tools*
Herramienta para crear y anexar a sistemas de ficheros squashfs
- *tftpd-hpa*
HPA's tftp server
- *xauth*
Herramienta de autenticación de X

La necesidad de estos programas se explica en el proceso de arranque de un thin client (Ver Anexo 10.1).

Una vez que el servidor ya está instalado, se procede a seleccionar una lista de paquetes educativos. Esta lista la he dividido en categorías, serán las siguientes:

Astronomía	Desarrollo	Diccionario	Física
Gnome	Gráficos	Idioma	Internet
Matemáticas	Música	Miscelánea	Multimedia
Niños	Oficina	Portatil	Herramientas

Cada una de las categorías tendrán los siguientes paquetes:

- **Astronomía**

Celestia	real-time visual space simulation
Gpredict	Satellite tracking program
Stellarium	real-time photo-realistic sky generator
Xplanet	planetary body renderer
Xplanet-images	imagery for xplanet

- **Desarrollo**

Bwbasic	Intérprete Basic de ByWater
Default-jdk	Última versión de openJDK
Eclipse	IDE para Java y extensible a otros lenguajes
Gambas2	IDE para Gambas y todas sus extensiones
Lazarus	IDE para Pascal
Python	Lenguaje de alto nivel python
Subversion	Sistema de control de versiones
Umbrello	Diagramas de flujo
Cpp-doc	Documentación del compilador GNU C++

- **Diccionario**

Myspell-es	Diccionario de español para Myspell
Ispanish	Diccionario de español para ispell
Wspanish	Spanish dictionary words for /usr/share/dic
myspell-ca	Diccionario de catalán de myspell

- **Física**

Planets	Gravitation simulation of planetary bodies
Xoscope	Digital oscilloscope

- **Gnome**

Gnome-accessibility-themes	Temas de accesibilidad del escritorio GNOME
Gnome-mag	Un ampliador de pantalla del escritorio GNOME
Gnome-orca	Lector de pantalla modificable mediante scripts
Gnome-themes	Temas oficiales del escritorio GNOME

Miguel Ángel Madrid Salinas

Gok	Teclado en pantalla de GNOME
-----	------------------------------

- Gráficos

Gimp	Programa para manipulación de imágenes GNU
Gimp-data-extras	Un juego adicional de brochas, paletas y gradientes para GIMP
Inkscape	Programa de dibujo vectorial

- Idioma

Iceweasel-l10n-es-es	Paquete de idioma para Iceweasel (Español)
Openoffice.org-help-es	Paquete de ayuda de openOffice.org (Español)
Openoffice.org-l10n-es	Paquete de idioma para openOffice.org (Español)
Gimp-help-es	Documentación de GIMP (español)

- Internet

Browser-plugin-gnash	Plugin para Mozilla y derivados del reproductor Flash de GNU
----------------------	--

- Matemáticas

Drgeo	Programa de geometría interactivo
Euler	Entorno de desarrollo matemático interactivo
Geogebra	Programa de matemáticas
Geomview	Programa de geometría interactivo
Gnuplot	Programa de plotting interactivo
Grace	Herramienta de plotting y gráficos XY
Graphmonkey	Calculadora grafica basada en GTK#
Graphthing	Herramienta para crear, manipular y estudiar grafos
Octave3.2	Lenguaje GNU para computaciones numéricas
Rocs	IDE para teoría de grafos
Xabacus	Simulación de la antigua calculadora (versión de X simple)

- Música

Audacity	Editor de audio
Denemo	gtk+ front end to GNU Lilypond
Gtick	Aplicación de metrónomo
Lmms	Linux Multimedia Studio
Rosegarden	Editor de música y secuenciador de MIDI y audio
Solfège	Entrenamiento de oído musical

- Miscelánea

Alsa-base	Utilidades para configurar y usar ALSA
Alsa-utils	Archivos de configuración del controlador de ALSA
Avahi-daemon	Demonio de mDNS/DNS-SD de Avahi
Default-jre	JRE de java versión openJRE
Gstreamer0.10-alsa	Complemento de GStreamer para ALSA
Gstreamer0.10-esd	Complemento de GStreamer para ESD
Gstreamer0.10-ffmpeg	Extensión de FFmpeg de GStreamer
Gstreamer0.10-fluendo-mp3	Complemento de GStreamer con el decodificador de mp3 de Fluendo
Gstreamer0.10-plugins-base	Complementos de GStreamer del conjunto «base»
Gstreamer0.10-plugins-good	Complementos de GStreamer del conjunto «bueno»
Gstreamer0.10-plugins-ugly	Complementos de GStreamer del conjunto «malo»

Miguel Ángel Madrid Salinas

Gstreamer0.10-pulseaudio	Complemento de GStreamer para Pulseaudio
menu	Genera menús de programas para todas las aplicaciones que sean de menú
Pulseaudio-esound-compatible	PulseAudio ESD compatibility layer

- Multimedia

Vlc	Reproductor multimedia
-----	------------------------

- Niños

Gcompris	Juegos educativos para los más pequeños
Gcompris-sound-es	Ficheros de audio en español para Gcompris

- Oficina

Openclipart-openoffice.org	Galería de <i>cliparts</i> para openOffice.org
----------------------------	--

- Portátil

Bluez-cups	Controlador de impresora Bluetooth para CUPS
Bluez-utils	Utilidades Bluetooth
Laptop-mode-tools	Herramientas para ahorro de energía basado en el estado Batería/CA
Usb-modeswitch	Herramienta de modo de conmutación para el control de dispositivos USB de tipo «flip flop»
Xserver-xorg-input-synaptics	Controlador de Synaptics TouchPad para el servidor X.Org

Una vez que está preparado el listado de paquetes, se guardan en un archivo llamado *listado_pkg* y se instalan todos de una sola vez con el siguiente comando:

```
# apt-get install -y `cat listado_pkg`
```

5.3 Fase 2: Configuración

Una vez que el servidor LTSP y todas sus dependencias estén instalados, es momento de configurarlo. Lo primero que habrá que hacer es configurar las interfaces de red, *eth0* y *eth1*.

Una vez identificadas las tarjetas se edita el archivo */etc/network/interfaces* dejándolo de esta manera:

```
auto lo
iface lo inet loopback

# inet interface
auto eth0
iface eth0 inet dhcp

# ltsp interface
auto eth1
iface eth1 inet static
address 10.0.1.254
netmask 255.255.255.0
network 10.0.1.0
broadcast 10.0.1.255
```

Las interfaces de red están configuradas para que una dé servicio de internet (*eth0*) y la otra servicio LTSP a los clientes (*eth1*). Aplicamos los cambios, reiniciando el servicio de red como root de la siguiente forma:

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

Una vez que la red vuelve a estar activa, se editan los archivos de configuración del servidor DHCP ubicados en */etc/ltsp/dhcpd.conf* y */etc/dhcp/dhcpd.conf* modificándolos de esta manera:

Miguel Ángel Madrid Salinas

```
authoritative;

subnet 10.0.1.0 netmask 255.255.255.0 {
    range 10.0.1.1 10.0.1.250;
    option domain-name "ltsp.org";
    option domain-name-servers 10.0.1.254;
    option broadcast-address 10.0.1.255;
    option routers 10.0.1.254;
    next-server 10.0.1.254;
    # get-lease-hostnames true;
    option subnet-mask 255.255.255.0;
    option root-path "/opt/ltsp/i386";
    if substring( option vendor-class-identifier, 0, 9 ) = "PXEClient" {
        filename "/ltsp/i386/pxelinux.0";
    } else {
        filename "/ltsp/i386/nbi.img";
    }
}
```

En este caso se le está indicando que sólo hay un servidor *DHCP* (*authoritative*). Su dirección IP coincidirá con la dirección de *eth1*, aquella que da servicio LTSP. Se le indica el rango de IPs a asignar (*range*), de la 10.0.1.1 a la 10.0.1.250. El enrutador (router y *next-server*) coincide con la IP de *eth1*, ya que sólo hay un servidor. La opción de ruta de origen para los equipos clientes (*option root-path*) hace referencia a la ruta donde los clientes cargarán el sistema. Esta imagen del sistema se construirá más adelante en */opt/ltsp/i386*, con el comando *ltsp-build-client*.

Se reinicia el servicio DHCP, comprobando que no salga ningún error.

```
#!/etc/init.d/isc-dhcp-server restart
```

Hay que indicar la interfaz de red que dará servicio DHCP.

```
# nano /etc/default/isc-dhcp-server
INTERFACES="eth1"
```

Se crean las imágenes en */opt/ltsp/i386* para los clientes, en este caso con arquitectura i386, la misma que el servidor, si fuera para 64bits o distinta del servidor deberíamos agregar el parámetro "--arch=[amd64 | i386]".

```
# ltsp-build-client
...
información: La instalación del cliente LTSP se instaló satisfactoriamente
```

Se edita el fichero */etc/exports* para *NFS*, dejando el directorio */opt/ltsp* a sólo lectura y que todos los usuarios puedan acceder a él. Añadiendo esta línea:

```
# echo '/opt/ltsp *(ro,no_root_squash,async)' >> /etc/exports
```

Se aplican los cambios realizados reiniciando el servidor NFS:

```
# invoke-rc.d nfs-kernel-server reload
```

Se edita el fichero */etc/default/tftpd-hpa*, para habilitar el servicio *tftp*, modificando la entrada *RUN_DAEMON* con el valor *yes*:

```
# nano /etc/default/tftpd-hpa
RUN_DAEMON="yes"
```

Se edita el fichero */etc/inetd.conf* comentando la línea que hace referencia al servicio *tftp*:

```
#tftp dgram udp wait root /usr/sbin/in.tftpd /usr/sbin/in.tftpd -s /var/lib/tftpboot
```

Una vez editado el archivo, se reinician los servicios *inetd* y *tftpd*.

Miguel Ángel Madrid Salinas

```
# invoke-rc.d openbsd-inetd restart
# invoke-rc.d tftpd-hpa restart
```

Ahora sólo queda crear un disco de arranque para aquellos clientes que no dispongan de tarjeta de red con soporte *PXE*. Desde <http://rom-o-matic.net/> se pueden descargar las imágenes, éstas pueden ser específicas para modelos particulares de tarjetas, aunque es preferible la versión *all-drivers*, ocupa 305KB y contiene todos los controladores disponibles. Una vez descargado, se graba en un disco.

```
# dd if=gpxe-1.0.1-gpxe.dsk of=/dev/fd0
```

Por último, se modifica el archivo de configuración de los clientes. Se llama *lts.conf* y está ubicado en */opt/lts/i386/etc/lts.conf*.

```
# lts.conf

[default]
LTSP_CONFIG=True      #
SOUND=True            # Se habilita el sonido
LOCALDEV=True         # Habilita los dispositivos locales, tales como el CDROM o USB
```

Se reinician estos servicios para asegurarnos de que los clientes puedan validarse en el servidor después de los cambios realizados.

```
# ltsupdate-sshkeys
# ltsupdate-image
```

Una vez que arranca el cliente, deberá aparecer una pantalla como ésta:



Ahora que ya está configurado el servidor y el cliente arranca correctamente, es momento para crear los grupos y los usuarios del sistema.

El criterio elegido será la de crear un grupo profesores y otro de alumnos. Cada uno de ellos tendrá un usuario, *profesor01* y *alumno01*.

```
# addgroup profesores
# addgroup alumnos
# useradd alumno01 --create-home --groups alumnos dialout fax cdrom floppy tape audio dip video plugdev fuse scanner
# useradd profesor01 --create-home --groups profesores alumnos alumno01 dialout fax cdrom floppy tape audio dip video plugdev fuse scanner
# chmod 700 /home/profesor01
# chmod 700 /home/alumno01
```

Como contraseña se pone la misma que el usuario. Una vez creados los usuarios y los grupos, se modifican los permisos en los directorios *home* de los usuarios, de modo que tengan acceso sólo los propietarios de los mismos. El usuario profesor debe poder acceder al directorio *home* del *alumno01*, por ello se le ha añadido el grupo *alumno01*.

6. Finalización

6.1 Creación del CD a partir del diseño

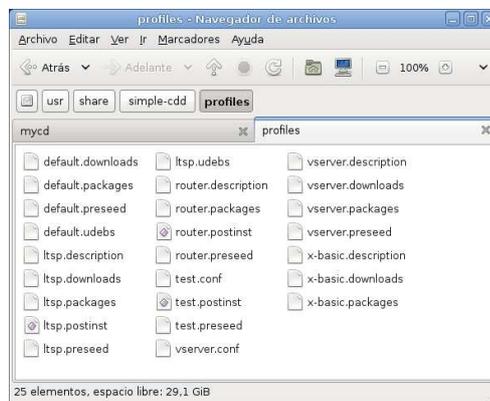
La herramienta que utilizaré para la creación del CD es *simple-cdd* y servirá para preconfigurar la instalación de Debian. Para ello crearé un directorio en el *home* de usuario:

```
$ mkdir ~/mycd ; cd ~/mycd
```

A continuación, como *root*, instalo el paquete *simple-cdd*:

```
# apt-get install simple-cdd
```

La documentación de *simple-cdd* está ubicada en */usr/share/simple-cdd/README*. Puedo leer que la herramienta se basa en perfiles. Cada perfil tiene un nombre y una extensión, el nombre hace referencia al perfil y la extensión a la tarea o modo de tratar el archivo por parte de *simple-cdd*. Hay perfiles de ejemplo en */usr/share/simple-cdd/profiles*.



Además de este paquete, es necesario el paquete de *udeb* de *debian-installer* para los perfiles de *simple-cdd*. Éste se encuentra en la rama inestable de *debian* y se descarga a *~/mycd*:

<http://packages.debian.org/unstable/simple-cdd-profiles>

El perfil *default.** se aplica por defecto, lo copio en *~/mycd/profiles*

<i>default.description:</i>	<i>Descripción del perfil.</i>
<i>default.downloads:</i>	<i>Listado de paquetes, un nombre por línea, que estarán en el CD pero no se instalarán.</i>
<i>default.packages:</i>	<i>Listado de paquetes, uno por línea, que se instalarán de manera predeterminada.</i>
<i>default.postinst:</i>	<i>Script que se ejecutará tras la instalación de paquetes.</i>
<i>default.preseed:</i>	<i>Archivo de respuestas automáticas para debconf, también conocido como preseed.</i>
<i>default.udebs</i>	<i>Listado de micro debs, en este caso solo habrá debs, por tanto, estará vacío.</i>

El archivo *default.packages* incluirá la lista de paquetes creada en el diseño del sistema, tanto el servidor LTSP como los paquetes educativos.

Otro archivo que es útil es el de configuración de *simple-cdd*. Hay uno de ejemplo en */usr/share/doc/simple-cdd/example/simple-cdd.conf*. Se copia éste también en *~/mycd* y se modifica de la siguiente manera:

```
# Profiles to include on the CD
profiles="" # Default profile is always included

# Idioma
```

Miguel Ángel Madrid Salinas

```
locale="es_ES"
language="Spanish"
country="Spain"

# Mirror tools
mirror_tools="wget reprepro"

# Mirror variables: coincide con server en sources.list
server="ftp.es.debian.org"
debian_mirror="http://$server/debian/"
wget_debian_mirror="ftp://$server/debian/"
rsync_debian_mirror="$server::debian"

# Mirror for security updates
# Expects security updates to be in dists/DEBIAN_DIST/updates
security_mirror="http://security.debian.org/"

# Extra mirror
# May only work when using reprepro as mirror tool
#debian_mirror_extra="http://example.com/debian/"

# Components to include
mirror_components="main contrib non-free"

# Generate a simple package repository on the CD with the debs cited
# Please insert full paths.
local_packages=""
```

El archivo *default.preseed* contendrá las respuestas automáticas del gestor *debconf* para una instalación desatendida. El instalador de Debian (*Debian Installer*) permite una instalación automática, sin preguntas, es lo que se conoce como *preseeding*. Consiste en un archivo de configuración que contiene las respuestas a cada una de las partes del instalador. Así pues, se podrá localizar la instalación, al idioma que queramos, el mapeado del teclado, configurar la red, etc.

Basándome en el [preseed de squeeze](#) modifiqué el archivo *default.preseed*. Este archivo servirá para crear el DVD más adelante. Entre otras tareas, lo que hará será localizar el idioma, mapear el teclado, configurar la primera tarjeta de red por dhcp, dará un nombre de equipo y de dominio; creará las particiones de forma automática o configurará la zona horaria.

```
#### Contents of the preconfiguration file (for squeeze)
### Localization
# Preseeding only locale sets language, country and locale.
d-i debian-installer/locale string es_ES

# Keyboard selection.
# Para mantener la configuración del núcleo y no salga el aviso al inicio (skip-config)
d-i console-tools/archs select skip-config
d-i console-keymaps-at/keymap select es
d-i keyboard-configuration/xkb-keymap select es

### Network configuration
# Presupone eth0 internet,
d-i netcfg/choose_interface select eth0
d-i netcfg/dhcp_timeout string 60

# hostname and domain
d-i netcfg/get_hostname string debian
d-i netcfg/get_domain string ltsp.org

### Mirror settings
# If you select ftp, the mirror/country string does not need to be set.
#d-i mirror/protocol string ftp
d-i mirror/country string manual
d-i mirror/http/hostname string ftp.es.debian.org
d-i mirror/http/directory string /debian
d-i mirror/http/proxy string

### Account setup
# Skip creation of a root account (normal user account will be able to
# use sudo).
```

Miguel Ángel Madrid Salinas

```
#d-i passwd/root-login boolean false
# Alternatively, to skip creation of a normal user account.
d-i passwd/make-user boolean false

# Root password, either in clear text
d-i passwd/root-password password l34Pme
d-i passwd/root-password-again password l34Pme
# or encrypted using an MD5 hash.

# To create a normal user account.
d-i passwd/user-fullname string profesor01
d-i passwd/username string profesor01
# Normal user's password, either in clear text
d-i passwd/user-password password profesor01
d-i passwd/user-password-again password profesor01
# or encrypted using an MD5 hash.
#d-i passwd/user-password-crypted password [MD5 hash]
# Create the first user with the specified UID instead of the default.
#d-i passwd/user-uid string 1010

# The user account will be added to some standard initial groups. To
# override that, use this.
d-i passwd/user-default-groups string audio cdrom video

### Clock and time zone setup
# Controls whether or not the hardware clock is set to UTC.
d-i clock-setup/utc boolean true

# You may set this to any valid setting for $TZ; see the contents of
# /usr/share/zoneinfo/ for valid values.
d-i time/zone string Europe/Madrid

# Controls whether to use NTP to set the clock during the install
d-i clock-setup/ntp boolean true
# NTP server to use. The default is almost always fine here.
#d-i clock-setup/ntp-server string ntp.example.com

### Partitioning
## Partitioning example
# If the system has free space you can choose to only partition that space.
# This is only honoured if partman-auto/method (below) is not set.
d-i partman-auto/init_automatically_partition select biggest_free

# Alternatively, you may specify a disk to partition. If the system has only
# one disk the installer will default to using that, but otherwise the device
# name must be given in traditional, non-devfs format (so e.g. /dev/hda or
# /dev/sda, and not e.g. /dev/discs/disc0/disc).
# For example, to use the first SCSI/SATA hard disk:
#d-i partman-auto/disk string /dev/sda
# In addition, you'll need to specify the method to use.
# The presently available methods are:
# - regular: use the usual partition types for your architecture
# - lvm: use LVM to partition the disk
# - crypto: use LVM within an encrypted partition
d-i partman-auto/method string lvm

# If one of the disks that are going to be automatically partitioned
# contains an old LVM configuration, the user will normally receive a
# warning. This can be preseeded away...
d-i partman-lvm/device_remove_lvm boolean true
# The same applies to pre-existing software RAID array:
d-i partman-md/device_remove_md boolean true
# And the same goes for the confirmation to write the lvm partitions.
d-i partman-lvm/confirm boolean true

# You can choose one of the three predefined partitioning recipes:
# - atomic: all files in one partition
# - home: separate /home partition
# - multi: separate /home, /usr, /var, and /tmp partitions
d-i partman-auto/choose_recipe select atomic

# This makes partman automatically partition without confirmation, provided
# that you told it what to do using one of the methods above.
```

Miguel Ángel Madrid Salinas

```
d-i partman-partitioning/confirm_write_new_label boolean true
d-i partman/choose_partition select finish
d-i partman/confirm boolean true
d-i partman/confirm_nooverwrite boolean true

# This makes partman automatically partition without confirmation.
#d-i partman-md/confirm boolean true
#d-i partman-partitioning/confirm_write_new_label boolean true
#d-i partman/choose_partition select finish
#d-i partman/confirm boolean true
#d-i partman/confirm_nooverwrite boolean true

## Controlling how partitions are mounted
# The default is to mount by UUID, but you can also choose "traditional" to
# use traditional device names, or "label" to try filesystem labels before
# falling back to UUIDs.
#d-i partman/mount_style select uuid

### Base system installation
# Configure APT to not install recommended packages by default. Use of this
# option can result in an incomplete system and should only be used by very
# experienced users.
#d-i base-installer/install-recommends boolean false

# Select the initramfs generator used to generate the initrd for 2.6 kernels.
#d-i base-installer/kernel/linux/initramfs-generators string initramfs-tools

# The kernel image (meta) package to be installed; "none" can be used if no
# kernel is to be installed.
#d-i base-installer/kernel/image string linux-image-2.6-486

### Apt setup
# You can choose to install non-free and contrib software.
#d-i apt-setup/non-free boolean true
#d-i apt-setup/contrib boolean true
# Uncomment this if you don't want to use a network mirror.
#d-i apt-setup/use_mirror boolean false
# Select which update services to use; define the mirrors to be used.
# Values shown below are the normal defaults.
#d-i apt-setup/services-select multiselect security, volatile
#d-i apt-setup/security_host string security.debian.org
#d-i apt-setup/volatile_host string volatile.debian.org

# Additional repositories, local[0-9] available
#d-i apt-setup/local0/repository string \
# http://local.server/debian stable main
#d-i apt-setup/local0/comment string local server
# Enable deb-src lines
#d-i apt-setup/local0/source boolean true
# URL to the public key of the local repository; you must provide a key or
# apt will complain about the unauthenticated repository and so the
# sources.list line will be left commented out
#d-i apt-setup/local0/key string http://local.server/key

# By default the installer requires that repositories be authenticated
# using a known gpg key. This setting can be used to disable that
# authentication. Warning: Insecure, not recommended.
#d-i debian-installer/allow_unauthenticated boolean true

### Package selection

# Instala el sistema estándar, el escritorio Gnome y herramientas para
# compatibilidad de portátiles.
tasksel tasksel/first multiselect standard, desktop, laptop

# Individual additional packages to install
#d-i pkgsel/include string openssh-server build-essential
# Se instalan de esta manera por lo explicado en la sección de errores encontrados
d-i pkgsel/include string celestia gpredict stellarium xplanet xplanet-images bwbasic default-jdk eclipse gambas2 lazarus
python subversion umbrello cpp-doc myspell-es ispanish wspanish myspell-ca planets xoscope gnome-mag gnome-orca
gnome-themes gok gnome-accessibility-themes gimp gimp-data-extras inkscape iceweasel-110n-es-es gimp-help-es browser-
plugin-gnash drgeo euler geogebra geomview gnuplot grace graphmonkey graphthing octave3.2 rocs xabacus audacity
denemo gtick lmms rosegarden solfege alsa-base alsa-utils avahi-daemon default-jre gstreamer0.10-alsa gstreamer0.10-esd
```

Miguel Ángel Madrid Salinas

```
gstreamer0.10-ffmpeg gstreamer0.10-fluendo-mp3 gstreamer0.10-plugins-base gstreamer0.10-plugins-good gstreamer0.10-
plugins-ugly gstreamer0.10-pulseaudio menu pulseaudio-esound-compat vlc gcompris gcompris-sound-es openclipart-
openoffice.org ltsp-server-standalone
# Whether to upgrade packages after debootstrap.
# Allowed values: none, safe-upgrade, full-upgrade
#d-i pkgsel/upgrade select none

# Some versions of the installer can report back on what software you have
# installed, and what software you use. The default is not to report back,
# but sending reports helps the project determine what software is most
# popular and include it on CDs.
popularity-contest popularity-contest/participate boolean false

### Finishing up the installation
# During installations from serial console, the regular virtual consoles
# (VT1-VT6) are normally disabled in /etc/inittab. Uncomment the next
# line to prevent this.
#d-i finish-install/keep-consoles boolean true

# Avoid that last message about the install being complete.
d-i finish-install/reboot_in_progress note

### Preseeding other packages
# Depending on what software you choose to install, or if things go wrong
# during the installation process, it's possible that other questions may
# be asked. You can preseed those too, of course. To get a list of every
# possible question that could be asked during an install, do an
# installation, and then run these commands:
# debconf-get-selections --installer > file
# debconf-get-selections >> file
#ltsp-client-builder ltsp-client-builder/run boolean true
#unattended-upgrades unattended-upgrades/enable_auto_updates boolean true
jackd2 jackd/tweak_rt_limits boolean false
tftpd-hpa tftpd-hpa/directory string /srv/tftp

# This command is run just before the install finishes, but when there is
# still a usable /target directory. You can chroot to /target and use it
# directly, or use the apt-install and in-target commands to easily install
# packages and run commands in the target system.
#d-i preseed/late_command string apt-install zsh; in-target chsh -s /bin/zsh
d-i preseed/late_command string cp -R /cdrom/simple-cdd /target; chroot /target; chmod +x default.postinst; sh
default.postinst > /dev/null 2>&1
```

Se crean las particiones LVM en el disco de forma automática. Se configura el archivo *preseed* con los parámetros de la instalación inicial (hostname, domain, etc). Se añaden también las líneas de respuesta a preguntas configuración de paquetes como tftpd o ltsp-server, construyendo el entorno del cliente. Se obtienen estas líneas por medio del siguiente comando, en el servidor ya configurado:

```
# debconf-get-selections | grep $Paquete
```

Cuando acabe con el proceso de instalación ejecutará un comando (*preseed/late_command*). El archivo *default.postinst* será el script que configurará el servidor una vez que los paquetes se hayan instalado. Se ejecutará dentro de un chroot, mientras el directorio */target* aún está disponible. Éste será su contenido:

```
#!/bin/sh

### Configure network
cat << EOF > /etc/network/interfaces
auto lo
iface lo inet loopback

# inet interface
auto eth0
iface eth0 inet dhcp

# ltsp interface
auto eth1
iface eth1 inet static
```

Miguel Ángel Madrid Salinas

```
address 10.0.1.254
netmask 255.255.255.0
network 10.0.1.0
broadcast 10.0.1.255
EOF
```

configure NFS

```
echo '/opt/ltsp *(ro,no_root_squash,async)' >> /etc/exports > /dev/null 2>&1
```

configure DHCP

```
echo 'include "/etc/ltsp/dhcpd.conf";' >> /etc/dhcp/dhcpd.conf
cat << EOF > /etc/ltsp/dhcpd.conf
authoritative;
```

```
subnet 10.0.1.0 netmask 255.255.255.0 {
    range 10.0.1.1 10.0.1.250;
    option domain-name "ltsp.org";
    option domain-name-servers 10.0.1.254;
    option broadcast-address 10.0.1.255;
    option routers 10.0.1.254;
    next-server 10.0.1.254;
    # get-lease-hostnames true;
    option subnet-mask 255.255.255.0;
    option root-path "/opt/ltsp/i386";
    if substring( option vendor-class-identifier, 0, 9 ) = "PXEClient" {
        filename "/ltsp/i386/pxelinux.0";
    } else {
        filename "/ltsp/i386/nbi.img";
    }
}
EOF
```

Configure isc-dhcp-server

```
sed 's/INTERFACES=""/INTERFACES="eth1"/g' /etc/default/isc-dhcp-server > /dev/null 2>&1
```

build client

```
ltsp-build-client > /dev/null 2>&1
```

Configure tftpd-hpa

```
echo RUN_DAEMON="yes" >> /etc/default/tftpd-hpa > /dev/null 2>&1
```

Modify inetd.conf

```
sed 's/tftp dgram udp wait root/#tftp dgram udp wait root/g' /etc/inetd.conf > /dev/null 2>&1
```

Configure lts.conf

```
echo SOUND=True >> /opt/ltsp/i386/etc/lts.conf > /dev/null 2>&1
echo LOCALDEV=True >> /opt/ltsp/i386/etc/lts.conf > /dev/null 2>&1
```

Add Groups

```
addgroup profesores > /dev/null 2>&1
addgroup alumnos > /dev/null 2>&1
```

Add Users

```
usuario_p=profesor01
usuario_a=alumno01
grupos_p="profesores,alumnos,audio,cdrom,dialout,dip,floppy,games,fuse,plugdev,scanner,tape,users,video"
grupos_a="alumnos,audio,cdrom,dialout,dip,floppy,games,fuse,plugdev,scanner,tape,users,video"
clave_p=$(openssl passwd -1 profesor01)
clave_a=$(openssl passwd -1 alumno01)
```

```
useradd -m -g "$usuario_p" -G "$grupos_p" -s /bin/bash -c "$usuario_p" -p "$clave_p" "$usuario_p" > /dev/null 2>&1
useradd -m -g "$usuario_a" -G "$grupos_a" -s /bin/bash -c "$usuario_a" -p "$clave_a" "$usuario_a" > /dev/null 2>&1
```

Add Users to Groups (preseed user)

```
adduser -q profesor01 profesores > /dev/null 2>&1
adduser -q profesor01 alumnos > /dev/null 2>&1
adduser -q alumno01 alumnos > /dev/null 2>&1
```

Update LTSP Client Environment

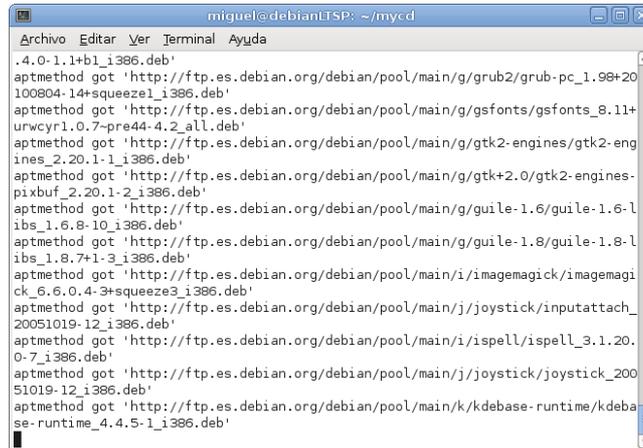
```
ltsp-update-sshkeys
ltsp-update-image
```

Miguel Ángel Madrid Salinas

Aplicará la configuración del servidor con los parámetros establecidos en el apartado de diseño del sistema. Una vez que se han creado estos archivos, se procede a ejecutar el siguiente comando con usuario normal, desde `~/mycd`:

```
$ build-simple-cdd --conf ./simple-cdd.conf --profiles-udeb-dist sid --local-packages ./simple-cdd-profiles_0.3.14_all.deb
```

El proceso puede durar bastante. Hará un *mirroring* del repositorio de debian dentro de `~/mycd/tmp` en función de los paquetes del listado `default.packages`, se incluirán también sus dependencias.

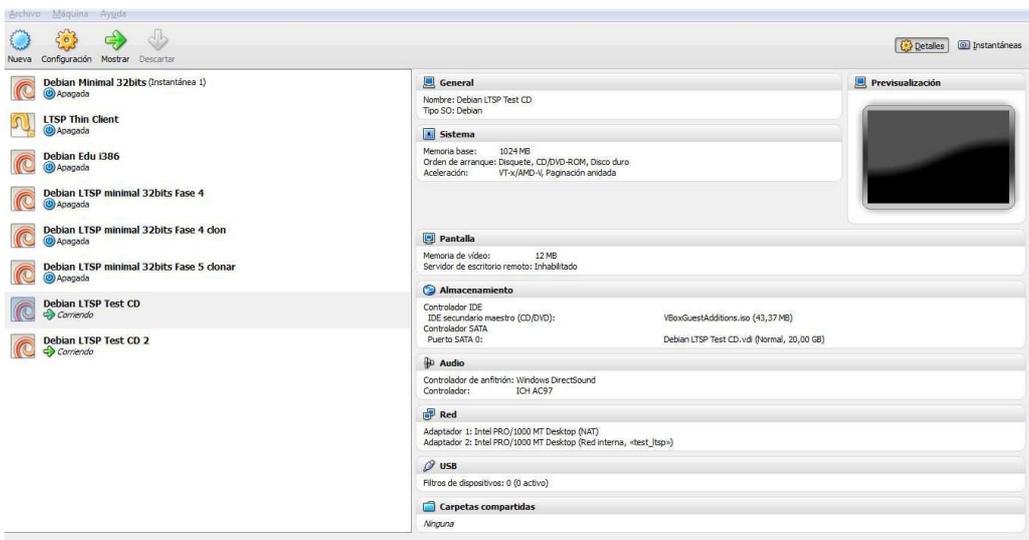


```
miguel@debianTSP: ~/mycd
Archivo Editar Ver Terminal Ayuda
.4.0-1.1+b1_i386.deb'
aptmethod got 'http://ftp.es.debian.org/debian/pool/main/g/grub2/grub-pc_1.98+20
100804-14+squeeze1_i386.deb'
aptmethod got 'http://ftp.es.debian.org/debian/pool/main/g/gsfonts/gsfonts_8.11+
urwcyr1.0.7-pre44-4.2_all.deb'
aptmethod got 'http://ftp.es.debian.org/debian/pool/main/g/gtk2-engines/gtk2-eng
ines_2.20.1-1_i386.deb'
aptmethod got 'http://ftp.es.debian.org/debian/pool/main/g/gtk+2.0/gtk2-engines-
pixbuf_2.20.1-2_i386.deb'
aptmethod got 'http://ftp.es.debian.org/debian/pool/main/g/guile-1.6/guile-1.6-l
ibs_1.6.8-10_i386.deb'
aptmethod got 'http://ftp.es.debian.org/debian/pool/main/g/guile-1.8/guile-1.8-l
ibs_1.8.7+1-3_i386.deb'
aptmethod got 'http://ftp.es.debian.org/debian/pool/main/i/imagemagick/imagemagi
ck_6.6.0.4-3+squeeze3_i386.deb'
aptmethod got 'http://ftp.es.debian.org/debian/pool/main/j/joystick/inputattach_
20051019-12_i386.deb'
aptmethod got 'http://ftp.es.debian.org/debian/pool/main/i/ispell/ispell_3.1.20.
0-7_i386.deb'
aptmethod got 'http://ftp.es.debian.org/debian/pool/main/j/joystick/joystick_200
51019-12_i386.deb'
aptmethod got 'http://ftp.es.debian.org/debian/pool/main/k/kdebase-runtime/kdeba
se-runtime_4.4.5-1_i386.deb'
```

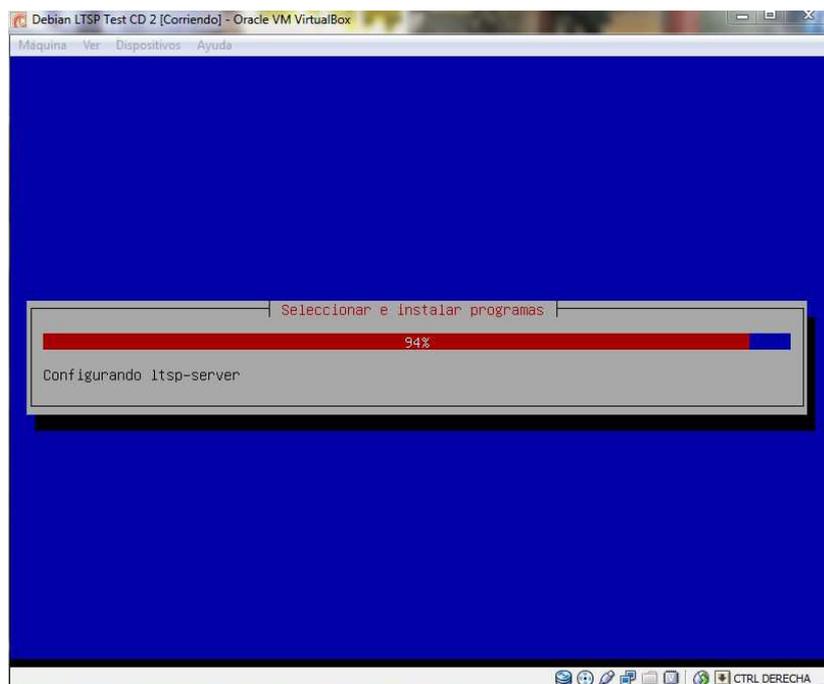
Cuando acabe habrá creado imágenes numeradas de CDs de instalación en el directorio `~/mycd/images`.

7 Pruebas

Las pruebas se realizarán sobre una máquina virtual, se instalará el CD creado en un disco virtual nuevo, con VirtualBox. Ésta se llamará "Debian LTSP Test CD".



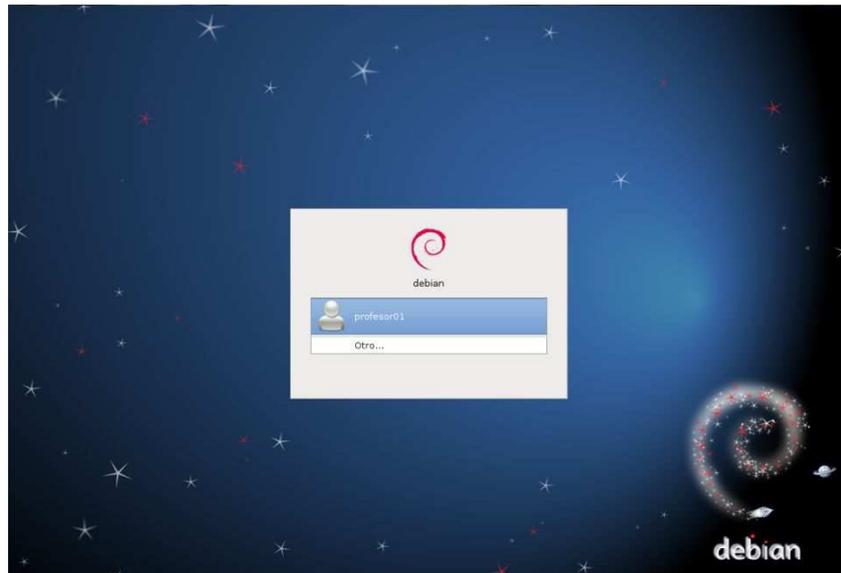
La máquina tendrá las mismas características que la máquina del diseño. Ésto significa que serán 1024 de RAM y dos tarjetas de red. Se añade el CD creado a la máquina y se procede a la instalación.



Miguel Ángel Madrid Salinas

Una vez que acabe el proceso, lo esperado es que haya instalado también el entorno del cliente y ejecutado el script *default.postinst* (última entrada del preseed).

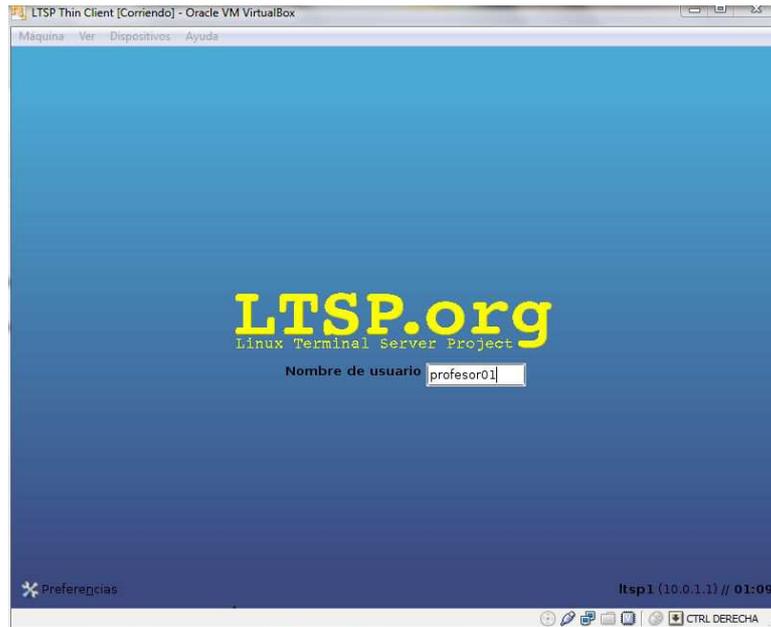
Al reiniciarse la máquina servidor se puede ver esta pantalla.



Por lo visto ha creado el usuario, Pulso [Ctrl] + [Alt] + [F1] para entrar en modo consola root y compruebo que todos los servicios estén configurados.

```
# cat /etc/network/interfaces
# cat /etc/default/isc-dhcp-server
# cat /etc/default/tftpd-hpa
# cat /etc/inetd.conf
# cat /opt/i386/i386/etc/ts.conf
```

Faltaba por configurar el DHCP, estaba en otra ruta. Una vez que está todo correcto, configuro la máquina virtual *thin client*, la red interna de la tarjeta eth1 debe ser test_ltsp; la inicio y entro como *profesor01*.



Se prueban aplicaciones de ofimática, de sonido, de internet, juegos, etc. Se comprueba con el monitor de sistema que el consumo no es excesivo. En la práctica puede variar, aunque es preferible seguir las recomendaciones de LTSP, un servidor con 4GB de RAM serán suficientes para albergar el consumo de unos 20 usuarios.

8 Errores encontrados y soluciones

8.1 Al crear el CD con simple-cdd fallan paquetes al descargarse

Al crear el cd con el perfil *debltsp* surge este error:

```
$ build-simple-cdd --conf ./simple-cdd.conf --profiles-udeb-dist sid --local-packages ./simple-cdd-profiles_0.3.14_all.deb
```

Error

```
...  
ERROR: missing required packages from profile default: less  
ERROR: missing required packages from profile default: simple-cdd-profiles  
WARNING: missing optional packages from profile default: grub popularity-contest type-handling syslinux-common syslinux  
ERROR: missing required packages from profile debltsp: ltsp-client xorg ltsp-server-standalone linux-image-486 pulseaudio gpredict  
stellarium xplanet xplanet-images lazarus subversion umbrello myspell-es myspell-ca ispanish wspanish planets xoscope gnome-  
accessibility-themes gnome-mag gnome-orca gnome-themes gok gimp gimp-data-extras gimp-help-es inkscape iceweasel-l10n-es-es  
openoffice.org-help-es openoffice.org-l10n-es gnuplot grace graphmonkey graphthing octave3.2 rocs xabacus gtick lmms rosegarden  
solfege gstreamer0.10-esd gstreamer0.10-ffmpeg gstreamer0.10-fluendo-mp3 gstreamer0.10-plugins-ugly gstreamer0.10-pulseaudio  
menu pulseaudio-esound-compatible vlc mplayer openclipart-openoffice.org laptop-mode-tools usb-modeswitch xserver-xorg-input-  
synaptics  
...
```

Este error está documentado en <http://wiki.debian.org/simple-cdd/howto>, en el apartado *Troubleshooting*. Las razones por las cuales estos paquetes no han sido añadidos están registradas en el log *tmp/cd-build/squeeze/sort_deps.i386.log*.

Tiene que ver con que no se han podido determinar algunas dependencias de estos paquetes. Por ejemplo, en el caso del paquete Lazarus, depende *lazarus-ide-gtk*, por medio de otra dependencia, *lazarus-ide*, es un paquete virtual que apunta a *lazarus-ide-gtk2*.

```
Lazarus (lazarus-ide Dep: ( OR lazarus-ide-gtk2 lazarus-ide-qt ))
```

Miguel Ángel Madrid Salinas

Por tanto, indican que se tienen que añadir estas dependencias de segundo nivel a `debltsp.downloads`. Ejecuto de nuevo el comando:

Error

```
Finished: 812 packages placed
including description: /home/miguel/mycd/profiles/debltsp.description
simple-cdd: extra files for simple-cdd
ERROR: missing required packages from profile default: less
WARNING: missing optional packages from profile default: popularity-contest type-handling syslinux-common syslinux
ERROR: missing required packages from profile debltsp: stellarium xplanet lazarus subversion umbrello myspell-es myspell-ca
wspanish planets xoscope openoffice.org-help-es openoffice.org-l10n-es octave3.2 rocs xabacus
WARNING: missing optional packages from profile debltsp: lazarus-ide-gtk2
```

Se añaden todos los paquetes instalados que hay en el sistema diseñado.

```
dpkg -l | awk {'print $2'} fs=" " >> profiles/default.packages
```

El programa sigue dando error porque no es capaz de verificar las claves GPG de los repositorios existentes. <http://lists.aliases.debian.org/pipermail/simple-cdd-devel/2011-February/000208.html>

Después de añadir estos paquetes a `*.packages`, se inicia de nuevo el comando `build-simple-cdd`, sale este error.

```
planets (= 0.1.13-11): FAILED
planets (= 0.1.13-11) depends on missing:
- ocaml-base-3.11.2
openoffice.org-writer (= 1:3.2.1-11+squeeze6): FAILED
openoffice.org-writer (= 1:3.2.1-11+squeeze6) depends on one of:
- openoffice.org-core (= 1:3.2.1-11+squeeze6)
openoffice.org-core (= 1:3.2.1-11+squeeze6) depends on one of:
- openoffice.org-common (= 1:3.2.1-11+squeeze6)
openoffice.org-common (= 1:3.2.1-11+squeeze6) depends on missing:
- openoffice.org-style-default
- openoffice.org-style
openoffice.org-help-es (= 1:3.2.1-11+squeeze6): FAILED
openoffice.org-help-es (= 1:3.2.1-11+squeeze6) depends on one of:
- openoffice.org-writer (= 1:3.2.1-11+squeeze6)
openoffice.org-writer (= 1:3.2.1-11+squeeze6) depends on one of:
- openoffice.org-core (= 1:3.2.1-11+squeeze6)
openoffice.org-core (= 1:3.2.1-11+squeeze6) depends on one of:
- openoffice.org-common (= 1:3.2.1-11+squeeze6)
openoffice.org-common (= 1:3.2.1-11+squeeze6) depends on missing:
- openoffice.org-style-default
- openoffice.org-style
openoffice.org-core (= 1:3.2.1-11+squeeze6): FAILED
openoffice.org-core (= 1:3.2.1-11+squeeze6) depends on one of:
- openoffice.org-common (= 1:3.2.1-11+squeeze6)
openoffice.org-common (= 1:3.2.1-11+squeeze6) depends on missing:
- openoffice.org-style-default
- openoffice.org-style
openoffice.org-common (= 1:3.2.1-11+squeeze6): FAILED
openoffice.org-common (= 1:3.2.1-11+squeeze6) depends on missing:
- openoffice.org-style-default
- openoffice.org-style
openoffice.org-base-core (= 1:3.2.1-11+squeeze6): FAILED
openoffice.org-base-core (= 1:3.2.1-11+squeeze6) depends on one of:
- openoffice.org-core (= 1:3.2.1-11+squeeze6)
openoffice.org-core (= 1:3.2.1-11+squeeze6) depends on one of:
- openoffice.org-common (= 1:3.2.1-11+squeeze6)
openoffice.org-common (= 1:3.2.1-11+squeeze6) depends on missing:
- openoffice.org-style-default
- openoffice.org-style
lazarus-ide (= 0.9.28.2-12): FAILED
lazarus-ide (= 0.9.28.2-12) depends on missing:
- lazarus-ide-gtk
- lazarus-ide-qt
lazarus (= 0.9.28.2-12): FAILED
lazarus (= 0.9.28.2-12) depends on one of:
```

Miguel Ángel Madrid Salinas

```
- lazarus-ide (= 0.9.28.2-12)
dnsutils (= 1:9.7.3.dfsg-1~squeeze4): FAILED
dnsutils (= 1:9.7.3.dfsg-1~squeeze4) depends on missing:
- libbind9-60 (= 1:9.7.3.dfsg-1~squeeze4)
find: «./simple-cdd-profiles_0.3.14_all.deb»: No existe el fichero o el directorio
ERROR: missing required packages from profile debtspp: <belocs-locales-data>
```

Se borra los directorio `tmp`, se cambia el valor del parámetro `--profiles-udeb-dist sid` por `squeeze` y se ejecuta de nuevo el comando `build-simple-cdd`.

```
build-simple-cdd --conf ./simple-cdd.conf --profiles-udeb-dist squeeze --local-packages ./simple-cdd-profiles_0.3.14_all.deb
```

Sólo con el perfil default, funciona.

Voy a añadir en grupos los paquetes en `debltsp.packages`, y a medida que fallen los sacaré de la lista para añadirlos uno a uno, más tarde.

Solución: A medida que se añaden paquetes, fallan algunos que anteriormente funcionaban, incluso he creado un script para ir añadiendo éstos paquetes de segundo nivel y sin éxito. Se añadirá la lista de paquetes al preseed de esta manera.

```
d-i pkgset/include string pkg1 pkg2 ...
```

Inconvenientes, estos paquetes se tienen que descargar de internet en lugar de iniciarse desde el CD.

9 Mejoras y conclusiones

El propósito del trabajo era crear una distribución personalizada desde cero, en este caso enfocada a la educación, que fuera libre, usable y fácil de mantener. Me hubiera gustado incluir más herramientas, realizar más pruebas, aunque quizás tendría que haberlo simplificado desde el principio. En todo caso, éstas son las posibles mejoras que incluiría.

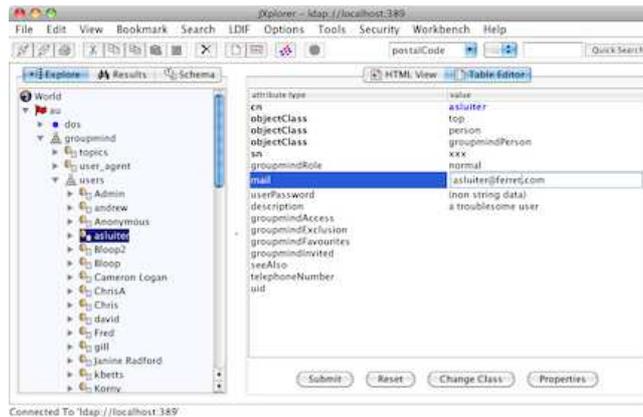
Configuración del sistema

Modificar archivos de configuración de sistema y de aplicaciones; de variables de entorno; el `skel` por defecto para los usuarios, esquemas, temas de escritorio personalizados, ejecutar archivos al arrancar el equipo, el usuario, establecer un sistema de cuotas para los usuarios controlando así el consumo de recursos, etc.

<code>/etc/xdg/autostart/</code>	Archivos que se ejecutan al iniciar sesión los usuarios
<code>/etc/rc[0..6].d</code>	Archivos que se ejecutan al apagarse el equipo (0), al arrancar en modo monousuario (1), en multiusuario (2-5) o cuando se reinicia (6).
<code>/etc/skel</code>	Archivos de preconfiguración de usuarios del sistema
<code>/etc/environment</code>	Archivo de configuración de variables globales (\$PATH)

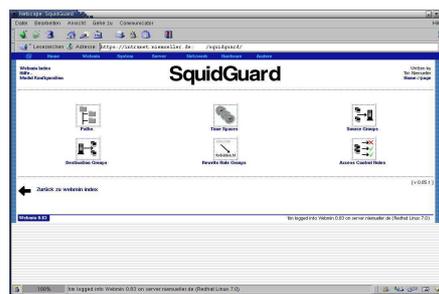
LDAP + Kerberos

Tener configurado el servidor con LDAP para la administración de usuarios y Kerberos para su autenticación en el sistema. Ésto permitiría administrarlo de una forma más eficiente e independiente. Se podría usar un frontend tipo [jxplorer](#) por parte de un usuario con permisos para ello (profesor p.ej).



Filtro de contenido Web

Otro aspecto importante sería controlar el contenido de las páginas web que se visiten. Ésto se puede realizar por medio de filtros de contenido, por ejemplo, con un servidor proxy cache como [Squid](#), y su complemento [squidguard](#), el cual permitiría establecer unas reglas de control de acceso basadas en unas determinadas listas o realizado por medio de expresiones regulares, etiquetas, etc.



Panel de configuración de SquidGuard desde Webmin.

Cortafuegos

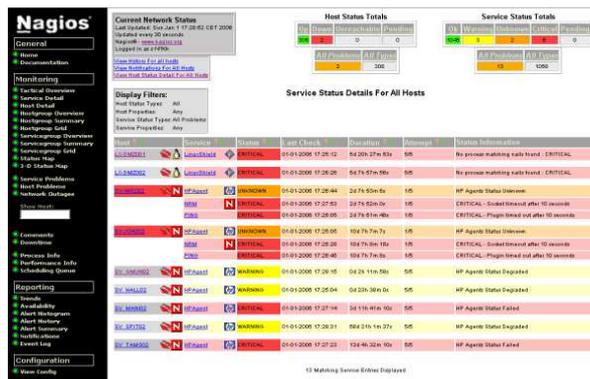
La mayoría de routers llevan un cortafuegos incorporado, aunque no estaría de más uno por software para controlar la seguridad del sistema. Siempre puede haber un corte de corriente, que reestablezca a valores de fábrica la configuración del router, ya que en muchos casos, por defecto no está activado. Usando iptables o algún frontend tipo ufw (Uncomplicated Firewall).

Acceso remoto

La mayoría de las IPs públicas de internet para particulares y empresas suelen ser dinámicas, pueden cambiar en cada reinicio del router. Una manera de acceder al servidor desde una red externa sería asignar una IP a la MAC de la tarjeta de red del servidor que va a dar servicio de internet. Esta IP se redirecciona al puerto 22 (SSH) desde la IP pública. Por otro lado, se crea una cuenta en servicios como [www.no-ip.com](#) para crear un dominio, tipo aula1.example.com. Para acceder, debemos configurar un cliente DNS en el servidor con los datos de esa cuenta para que vaya actualizándose la IP pública cada cierto tiempo. Todo esto permitiría administrar remotamente el servidor accediendo por `ssh` a esa dirección, aula1.example.com (por ejemplo, mediante un cliente `ssh` como [Putty](#)), incluso se podría realizar gráficamente usando un cliente NX, antes habría que instalar [FreeNX](#) en el servidor.

Monitorización

Además, el estado del servidor podría monitorizarse, si es necesario, mediante una solución como [Nagios](#), como sistema de alertas. Se podría realizar remotamente mediante SSH. Comprobar el estado de los servicios, logs del sistema, etc. Se podrían crear alertas personalizadas para elementos propios del LTSP.



Panel de Nagios

RAID

Habilitar el Preseed de debian-installer para RAID, antes habría que comprobar el estado de los discos:

```
# This command is run immediately before the partitioner starts. It may be
# useful to apply dynamic partitioner preseeding that depends on the state
# of the disks (which may not be visible when preseed/early_command runs).
#d-i partman/early_command \
# string debconf-set partman-auto/disk "$(list-devices disk | head -n1)"
```

Este comando se ejecutaría antes del particionado (early_command) y ayudaría a particionar el disco de una manera más precisa, conociendo a qué archivo disco hace referencia.

```
## Partitioning using RAID
# The method should be set to "raid".
#d-i partman-auto/method string raid
# Specify the disks to be partitioned. They will all get the same layout,
# so this will only work if the disks are the same size.
#d-i partman-auto/disk string /dev/sda /dev/sdb

# Next you need to specify the physical partitions that will be used.
#d-i partman-auto/expert_recipe string \
# multiraid ::
# 1000 5000 4000 raid \
# $primary{ } method{ raid } \
# . \
# 64 512 300% raid \
# method{ raid } \
# . \
# 500 10000 1000000000 raid \
# method{ raid } \
# .

# Last you need to specify how the previously defined partitions will be
# used in the RAID setup. Remember to use the correct partition numbers
# for logical partitions. RAID levels 0, 1, 5, 6 and 10 are supported;
# devices are separated using "#".
# Parameters are:
# <raidtype> <devcount> <sparecount> <fstype> <mountpoint> \
# <devices> <sparedevices>

#d-i partman-auto-raid/recipe string \
# 1 2 0 ext3 / \
# /dev/sda1#/dev/sdb1 \
# . \
# 1 2 0 swap - \
# /dev/sda5#/dev/sdb5 \
# . \
# 0 2 0 ext3 /home \
# /dev/sda6#/dev/sdb6 \
# .

# For additional information see the file partman-auto-raid-recipe.txt
```

Miguel Ángel Madrid Salinas

```
# included in the 'debian-installer' package or available from D-I source  
# repository.
```

Nota: Este mismo procedimiento se podría aplicar para determinar las tarjetas de red.

Copias de seguridad

Sería posible utilizar alguna herramienta tipo [rsync](#) para realizar copias de seguridad incrementales de manera periódica. Por ejemplo, se podrían crear tareas mediante *cron* para que las realizase por medio de una conexión *ssh*, a otro equipo o unidad.

10 Anexos

10.1 Proceso de arranque de un cliente con LTSP

El proceso de arranque de un cliente a través de un servidor LTSP es el siguiente:

1. Se carga el kernel linux en la memoria RAM del cliente. Este proceso puede hacerse de diferentes modos: PXE, etherboot o Yaboot. El método más usual y sencillo es PXE, aunque si la tarjeta no lo soporta podemos usar los otros.
2. Una vez que el kernel se ha cargado en la memoria RAM, se ejecutará.
3. El kernel inicializará el sistema reconociendo los periféricos que se vaya encontrando.
4. Durante el proceso de carga del kernel, una imagen llamada initramfs será cargada también en memoria RAM.
5. Cuando se haya completado la carga del kernel ejecutará el script init alojado en la raíz de initramfs.
6. El script init monta los directorios /proc y /sys, inicia udev para detectar hardware, empezando por la tarjeta de red. También crea una zona en la memoria RAM del cliente para almacenaje local para ir escribiendo en ella.
7. Se configura la interfaz de red loopback con IP 127.0.0.1.
8. Se carga un cliente DHCP reducido que llama al comando ipconfig para consultar al servidor DHCP.
9. Cuando ipconfig obtiene respuesta del servidor, se configura la interfaz de red.

Miguel Ángel Madrid Salinas

10. El script init montará un nuevo sistema de ficheros, esta vez mediante NBD o NFS. Las imágenes están en /opt/ltsp/images o /opt/ltsp/<arch>, donde <arch> es la arquitectura utilizada, i386, amd64, etc.
11. Una vez que el nuevo sistema de ficheros está montado, se ejecuta de nuevo el script init, pero esta vez desde /sbin/init.
12. init leerá el directorio /etc/event.d y empezará configurando el entorno del cliente. Desde ese lugar, upstart empezará ejecutando los comandos que hay en /etc/rcS.d.
13. Ejecutará el comando ltsp-client-setup el cual configurará muchas funcionalidades del entorno del cliente, como los dispositivos locales, carga de módulos, etc.
14. Una vez terminado, init ejecutará los comandos que hay en /etc/rc2.d.
15. Uno de los comandos que hay en ese directorio es el ltsp-client-core, que se cargará mientras el cliente está iniciándose.
16. Se leerá el fichero lts.conf, y todos los parámetros específicos para un cliente serán añadidos como variables al script ltsp-client-core.
17. Si está configurado el sonido, se iniciará el daemon de pulseaudio, que permitirá una conexión de audio remota al servidor y reproducirlo en el cliente.
18. Si el cliente tiene habilitado el soporte para dispositivos locales, se iniciará el programa ltspsfd para permitir al servidor leer desde estos dispositivos (usb, cdrom...).
19. Se ejecutarán las sesiones de pantalla (screen sessions) que estén definidas en lts.conf. Si no se especifica, el script de pantalla se ejecuta en SCREEN_07 con LDM (LTSP Display Manager), con su valor ldmdm.
20. Por defecto, leerá la configuración en /etc/X11/xorg.conf, e iniciará esta configuración.
21. El servidor X iniciará un túnel SSH cifrado hacia el servidor.
22. El usuario puede validarse para obtener una sesión en el servidor.

A partir de este momento, todas las operaciones como la ejecución de aplicaciones serán administradas desde el servidor, en lugar de en el cliente. El servidor LTSP transfiere por red toda la información gráfica. Por ejemplo, si hay veinte clientes ejecutando el programa openOffice.org, el servidor lo cargará una sola vez y creará instancias para cada usuario que lo solicite.

10.2 Creación masiva de usuarios

A continuación muestro un script que crearía usuarios de forma masiva en el equipo. Crearía 3 usuarios de tipo profesor (profesor001..003) y 20 de tipo alumno (alumno001..020), con la contraseña la misma que el usuario.

```
# setusers profesor 3  
# setusers alumno 20
```

```
#!/bin/bash  
## Añadir un número determinado de usuarios que tienen en común el nombre.  
## setusers {nombre} {cantidad de usuarios}  
encargado="profesor"  
user="alumno"  
usuario="$1"
```

Miguel Ángel Madrid Salinas

```
nUsuarios=$(awk -F":" '{print $1}' /etc/passwd | grep "$1" | wc -l)
nEncargados=$(awk -F":" '{print $1}' /etc/passwd | grep "$encargado" | wc -l)
min=$((nUsuarios + 1))
max="$2"
n=0
grupos="audio,cdrom,dialout,dip,floppy,games,lpadm,plugdev,scanner,tape,users,video"
a="$usuario"
e="$usuario"

setUsuario ()
{
    echo Creando "$1"
    groupadd "$1"
    clave=$(openssl passwd -1 $1)
    useradd -m -g "$1" -G "$grupos" -s /bin/bash -c "$1" -p "$clave" "$1"
    chmod 750 /home/"$1"
    # chown -Rf "$1"."$1" /home/"$1"
}

getEncargado ()
{
    echo Añadiendo profesores a grupo de alumnos
    for j in `awk -F":" '{print $1}' /etc/passwd | grep "$encargado"; do
        for k in `awk -F":" '{print $1}' /etc/passwd | grep "$usuario"; do
            echo usuario "$j" en grupo "$k"
            adduser "$j" "$k" > /dev/null
        done
    done
}

if [ ""id -un` != "root" ]; then
    echo 'Has de ejecutar el script como root'
    exit
elif [ "$#" != 2 ] || [ "$2" -le 0 ];then
    echo Uso: $0 {nombre} {cantidad de usuarios}
    exit
elif [ "$nEncargados" -le 0 ] && [ "$1" != "$encargado" ]; then
    echo No se ha creado todavía ningún usuario "$encargado".
    echo Uso: $0 "$encargado" {cantidad de usuarios}
    exit
fi

echo Se crearán "$2" "$usuario"s, hay "$nUsuarios" creados con el mismo nombre como prefijo.

while [ "$n" != "$max" ]
do
    if [ "$min" -lt 10 ];then
        a="$usuario"00"$min"
    elif [ "$min" -ge 10 ] && [ "$min" -lt 100 ];then
        a="$usuario"0"$min"
    else
        a="$usuario"$min"
    fi
    setUsuario "$a"
    let "n+=1"
    let "min+=1"
done

if [ "$usuario" == "$luser" ]; then
    getEncargado
else
    echo 'Tendrás que cambiar el password a "$usuario": passwd "$usuario"'
fi

echo Done
```

11 Bibliografía

<http://www.debian.org>

<http://ltsp.org>

<http://wiki.debian.org>

<http://wikipedia.org>

<http://www.debian.org/releases/stable/s390/apbs04.html.es>