

PBX Asterisk embebida

David Temes López (dtemes@mailtovoice.com)
Carlos Temes Caneda (ctemes@meutrives.com)

Mayo de 2008

Objetivos

El objetivo de este proyecto es montar una centralita telefónica basada en Asterisk que cumpla con los siguientes requisitos:

- Carecer de partes móviles
- Bajo consumo
- Capacidad de interactuar con líneas analógicas convencionales

Material utilizado

- Mini PC Alix3c2 y Alix3c3
- Tarjeta de telefonía OpenvVox A400M con 1 FXO y 1 FXS
- Tarjeta Compact Flash de 1Gb
- Caja, separadores y tornillería para montaje del prototipo

Las placas Alix han sido seleccionadas por los buenos resultados obtenidos en anteriores ocasiones al usarlas para el montaje de firewalls con el software m0n0wall. La diferencia entre la Alix3c2 y Alix3C3 es que la última incorpora salida VGA que será de utilizada para depurar posibles fallos. Será la Alix3c2 la que finalmente sea montada en el prototipo 2.

Además de los elementos citados se ha utilizado un ordenador, un lector de tarjetas compact flash y herramientas varias, como soldador y destornilladores.

Preparación del software

En el apartado software se ha escogido Debian Etch como base. La instalación no se realizará directamente sobre el soporte CF, ni se realizará sobre el prototipo, sino que se preparará en un ordenador de escritorio, el mismo en el que escribo ahora, con la distribución de Linux Uuntu 8.04.

Esta parte del documento no pretende ser un tutorial paso a paso , si no que es un resumen de alto nivel de las tareas realizadas haciendo hincapié en aquellos aspectos más complicados o curiosos.

Parte I

1. Preparación del filesystem: Con ayuda de *dd* y *mke2fs* se ha preparado una imagen de nuestro sistema de 1 Gb de capacidad, que albergará el root filesystem. A esta imagen se ha llamado root.alix-1024. Se ha creado otra imagen auxiliar de similar tamaño llamada root.alix-src que en su momento se utilizará para descargar y compilar las fuentes de Asterisk.
2. Carga del sistema operativo: Una vez listo el filesystem se ha montado en */mnt* (*mount -o*

- loop root.alix-1024 /mnt*) y se ha ejecutado *debootstrap* sobre el para cargar una imagen de Debian Etch. Tras la carga se ha hecho un *chroot /mnt* y se han retocado algunos ficheros de configuración, como */etc/hosts*, */etc/resolv.conf*, */etc/fstab*, configuración de la red, etc.
3. Adecuación del kernel: Por las características de la plataforma destino, se han instalado vía *apt-get* los binarios y cabeceras del kernel 2.6.18-6-486. Con *mkinitramfs* se ha creado un ramdisk apropiado para dicho kernel.
 4. Descarga del software: En este momento entra en juego nuestro segundo filesystem, que montaremos en */mnt/usr/src* de forma que será visto una vez estamos en “chroot” como */usr/src*. Aquí se ha descargado el código fuente de Asterisk (*zaptel*, *libpri*, *asterisk* y *asterisk-addons*) en su última versión y se han descomprimido los tar. De nuevo vía *apt-get* se han instalado algunas dependencias y herramientas de compilación. Entre otras cosas se ha instalado el gestor de arranque *Grub*.
 5. Compilación: A la hora de compilar se ha seguido el orden *zaptel*, *libpri*, *asterisk* y *asterisk-addons* y lo único a tener en cuenta es que el target no será nuestro propio ordenador, por lo que al *./configure* le pasaremos la opción *--host* para indicar que queremos compilar para un “486”. Tras compilar cada elemento se ha hecho un *make install*, *make config* y *make samples* según fuera necesario.
 6. Pruebas iniciales: Se han montado con opción *bind* los filesystem *sys*, *dev* y *proc* en nuestro filesystem en chroot. Tras esto el asterisk arranca e incluso se ha configurado un softphone y se ha probado que se registra correctamente y un dialplan sencillo.
 7. Post instalación de la imagen: En nuestro sistema no podían faltar las fantásticas voces en Español que se pueden obtener del blog Voipnovatos (<http://www.voipnovatos.es>) así que para comenzar las descargamos en formato *alaw* y *gsm*.

Bueno, ya tenemos nuestra imagen lista. Lo bueno de tener todo un filesystem en un único fichero es que es muy fácil hacer copias de seguridad y trabajar con diferentes versiones del mismo, así que llegados a este punto no está de más hacer una copia.

Parte II

Llegados a este punto hay que cargar todo el software que hemos preparado en una tarjeta Compact Flash, que es lo que “come” la Alix, así que manos a la obra.

1. Particionado: A la hora de particionar la CF se ha buscado ser práctico, y se ha optado por hacer dos particiones, la primera, de unos 20Mb se montará en */boot* y queda claro cual será su finalidad. La segunda, con el resto de espacio disponible será la raíz del sistema. Así es como quedó la tabla de particiones:

```
Command (m for help): p
Disk /dev/hda: 1031 MB, 1031675904 bytes
255 heads, 63 sectors/track, 125 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hda1    *           2           4       24097+   83  Linux
/dev/hda2                5          125      971932+   83  Linux

Command (m for help):
```

2. Copia de ficheros: Una vez preparadas las particiones se han montado en */mnt1* y */mnt2* y se han copiado los ficheros correspondientes a cada una desde la imagen en */mnt* empleando *cp* y *sync*. Estos son los contenidos de */etc/fstab*:

```
/dev/hda2  /      ext2  default s  1    1
/dev/hda1  /boot ext2  default s  1    1
```

3. Instalación del gestor de arranque: He de decir que la tarjeta CF de uso fotográfico adquirida con urgencia en una casa de fotografía a un precio desorbitado me salió rebelde. Han sido varios los intentos fallidos de instalar directamente Centos, Debian y Ubuntu. En todos los casos tras una instalación sumamente lenta el resultado fue la imposibilidad de hacer boot desde dicha memoria, generalmente con un error 17 en el “second stage” del Grub. En esta ocasión no me daría por vencido fácilmente y tras unos cuantos ciclos de ensayo-error di con la combinación ganadora. Con la CF en el lector desde el equipo anfitrión se usa grub para instalar en ella el gestor de arranque. Estos han sido los comandos principales de grub:

```
device (hd0) /dev/sdc          ; mapea la CF como (hd0)
cat (hd0)/etc/hostname        ; compruebo que trabajo sobre la
unidad                          ; correcta
root(hd0, 0)                   ; indica que se hará boot desde la
                                ; primera partición
setup (hd0)                     ; instala grub
```

Si no hay ningún mensaje de error ya estamos listos para arrancar desde nuestra CF.

Parte III

El equipo ya arranca, pero hay una serie de problemas:

1. No hay red: Monitorizando el proceso de boot con gran entusiasmo y emoción veo entre la multitud de líneas como se detectan dispositivos...NET.....HDA...y finalmente llego al esperado prompt para hacer login, sin embargo no tengo teclado usb, por lo que tendré que entrar por red. Miro en mi servidor DHCP y no veo que haya entregado ip a ningún equipo...hago un par de pruebas miro cableado....nada, no hay red. Apago a lo bestia y a revisar ficheros. Claro que no hay red, el fichero `/etc/network/interfaces` está vacío!!! Una vez configurado vuelvo a intentarlo y varios ciclos de reboot por tener que hacer chequeos del filesystem, pero finalmente arrancó y pude entrar por red.
2. ¿Dónde están mis dispositivos zap? Lo primero que pensé es que al haber compilado `zaptel` y hecho el `make install` en el anfitrión los dispositivos me los ha creado en mi ordenador (por aquello de tener el `/dev` montado con opción `bind` en la imagen del filesystem). Creo los dispositivos a mano con `MAKEDEV` y el comando `zap show status` me muestra el fxo y fxs correctamente. Hago las primeras llamadas de prueba entrantes, salientes, conferencias...todo funciona. Sin embargo tras reiniciar el equipo de nuevo los dispositivos zap han desaparecido. Observo con `mount` que el tipo de filesystem montado en `/dev` es `udev`, sin embargo no tengo `udevinfo` ni otros ejecutables asociados...

```
apt-get install udev
reboot
```

Voila!!! todo funciona perfectamente.

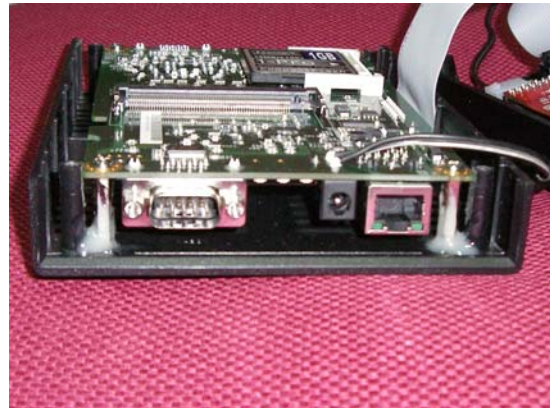
Preparación del hardware

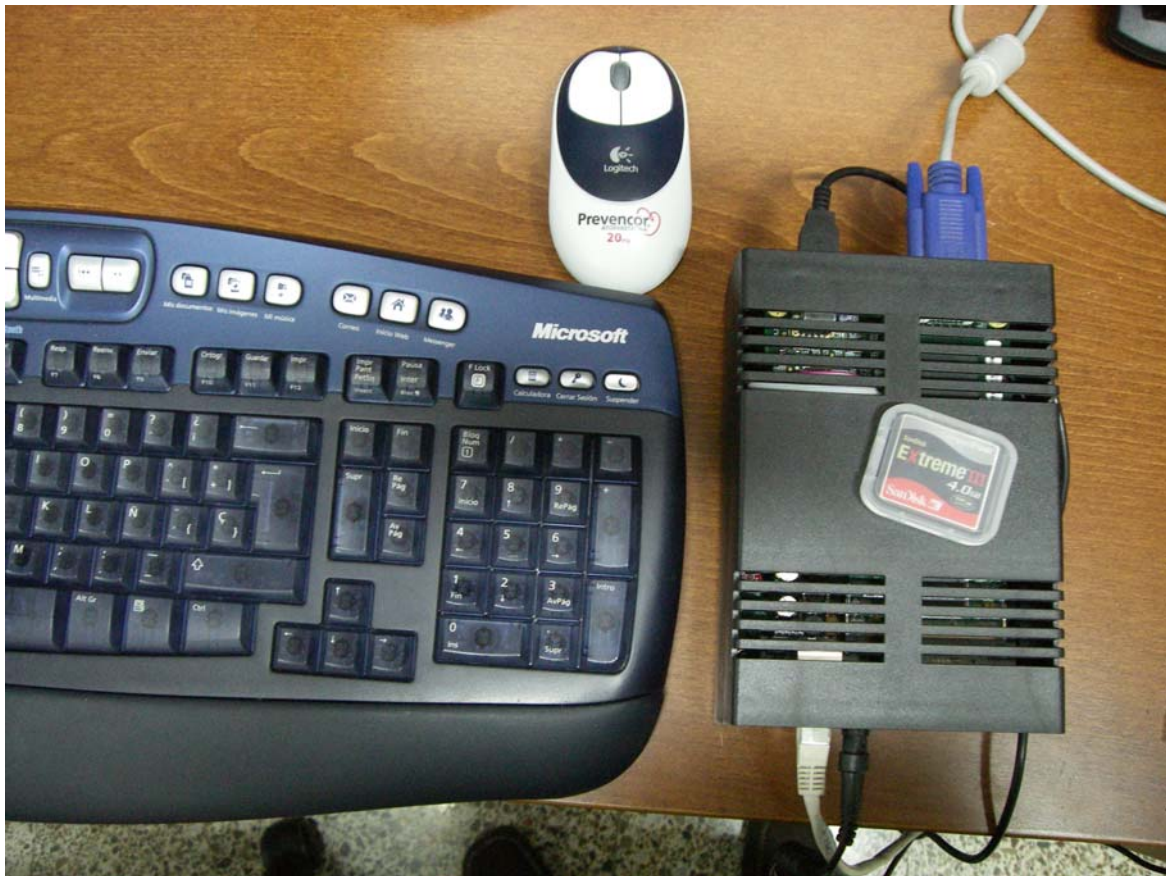
El dispositivo funciona correctamente pero está al aire sobre su propia caja de cartón. Va siendo hora de preparar una caja adecuada y poner un poco de orden antes de que nos echen de casa.

Nos fuimos de visita a una tienda de electrónica y salimos con una bolsita que contenía:

- Caja de plástico
- Separadores de varias longitudes
- Tornillos (estaban justo al lado de los separadores pero no eran de la medida adecuada, grrr!!!)

Aquí algunas fotos del montaje en diferentes fases:





Post instalación y mejoras

Tras montar y comprobar el funcionamiento se han hecho una serie de mejoras y adecuaciones del software entre las que cabe destacar la creación de un contexto con extensiones para apagar o resetear el sistema, y sobre todo, la modificación de */etc/fstab* para montar los ficheros en modo solo lectura utilizando *tmpfs* en aquellos puntos donde la escritura se hace necesaria, como */tmp* y */var* y sus subdirectorios. Con estos cambios ya podemos quitar la corriente sin miedo al posible deterioro de datos que podría dejar nuestro appliance inservible. Ahora queda pendiente corregir algún problema derivado del montaje en modo read-only y el estudio de otros sistemas de ficheros más apropiados para sistemas embebidos.

Datos de funcionamiento

Utilizando el codec alaw hemos metido 5 llamadas (2 por canales zap y 3 clientes SIP) en una teleconferencia y la CPU se ponía puntualmente al 11%, aunque durante picos muy cortos para bajar después y mantenerse por debajo del 3%. Me atrevería a decir que sin hacer transcoding se podrían gestionar sin problemas hasta 10 llamadas. Con algunas combinaciones de codecs sin embargo los tiempos de traducción se disparan y el escenario no sería tan favorable. En concreto es en caso de utilizar g729 (Con la versión “libre” que corre por ahí compilada para Pentium III)

Translation times between formats (in milliseconds) for one second of data														
Source Format (Rows) Destination Format (Columns)														
	g723	gsm	ulaw	alaw	g726aal2	adpcm	slin	lpc10	g729	speex	ilbc	g726	g722	
g723	-	-	-	-	-	-	-	-	-	-	-	-	-	-
gsm	-	-	8	8	20	8	7	43	146	-	-	20	-	-
ulaw	-	23	-	1	14	2	1	37	140	-	-	14	-	-
alaw	-	23	1	-	14	2	1	37	140	-	-	14	-	-
g726aal2	-	34	13	13	-	13	12	48	151	-	-	1	-	-
adpcm	-	23	2	2	14	-	1	37	140	-	-	14	-	-
slin	-	22	1	1	13	1	-	36	139	-	-	13	-	-
lpc10	-	45	24	24	36	24	23	-	162	-	-	36	-	-
g729	-	56	35	35	47	35	34	70	-	-	-	47	-	-
speex	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ilbc	-	-	-	-	-	-	-	-	-	-	-	-	-	-
g726	-	34	13	13	1	13	12	48	151	-	-	-	-	-
g722	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Proximos pasos

A modo de ensayo se ha instalado Freepbx en una segunda tarjeta CF (corriendo sobre Lighttpd en lugar de Apache). El funcionamiento es aceptable, el uso de CPU y memoria es mínimo si bien la capacidad de almacenamiento está a límite aunque se podría recuperar algo de espacio eliminando documentación y algunas librerías y ejecutables no requeridos.

Queda pendiente hacer pruebas más extensivas intentando buscar el límite del hardware y ver qué ocurre si se alcanza, y sobre todo nos gustaría despejar una incógnita: ¿sería capaz la Alix de manejar una quad-bri y aguantar la carga de 8 llamadas? En caso afirmativo sería una opción a considerar para la fabricación de appliances Asterisk.

Enlaces de interés

Este proyecto se ha podido llevar a cabo en gran medida gracias a la ayuda de otras personas que han decidido compartir sus conocimientos y su obra en Internet. Aquí pongo algunos enlaces de interés en los que he podido encontrar valiosa información:

[VoIP para novatos - Sólo de Voz sobre IP vive el hombre](#)

[Making a bootable backup Debian system disk](#)

<http://peter.wm.sk/p/?p=20>

<http://www.pcengines.ch/alix3c3.htm>

<http://www.openvox.com.cn/downloadsFile/A400M%20Datasheet.pdf>



Esta obra está bajo una licencia Reconocimiento 2.5 España de Creative Commons. Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by/2.5/es/> o envíe una carta a Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.