

Exploración del gestor de arranque de nueva generación GRUB

REBOOT

El tan esperado GRUB 2 trae al gestor de arranque GRUB un nuevo aspecto y un montón de nuevas características. **POR ANDREAS BOHLE**



Lars Lieber, Fotolia

El gestor de arranque GRUB apareció en escena hace varios años como una alternativa al popular LILO (Linux Loader), y desde entonces se ha convertido en una herramienta estándar para la carga y puesta en marcha del sistema Linux. Muchos usuarios finales asocian GRUB con el menú de arranque que aparece durante el proceso de inicio en equipos con una configuración de arranque múltiple. Sin embargo, aunque no interactúes directamente con GRUB, éste sigue funcionando en la trastienda cuando busca y carga el kernel para poner en marcha el sistema.

El gestor de arranque GNU GRUB estándar está basado en el GRUB original (Grand Unified Loader) creado por Erich Stefan Boleyn. GRUB ha servido bien a la comunidad durante años, pero desde hace algún tiempo, los desarrolladores vienen notando sus achaques. Y, a pesar de que el código base ha sido mejorado a lo largo del tiempo, al programa le faltan aún algunas funcionalidades importantes que podrían ser de utilidad para la próxima generación de ordenadores.

El proyecto PUPA (Preliminary Universal Programming Architecture for GNU GRUB) [1] fue una iniciativa de investi-

gación (patrocinada por la Agencia de Promoción de la Información y la Tecnología de Japón) cuya finalidad era volver a reorientar y revitalizar GRUB. Los desarrolladores de GNU adoptaron esta iniciativa y lanzaron una nueva versión de GRUB basada en PUPA.

GRUB 2 [2] aparece en las nuevas versiones de las principales distribuciones Linux. (Si aún no ha salido en la tuya, probablemente aparezca en la siguiente versión). Según palabras de los desarrolladores del proyecto GNU GRUB 2 "...ha sido reescrito desde cero para mejorar así toda su modularidad y portabilidad".

Los objetivos perseguidos para GRUB 2 son proporcionar:

- soporte para scripts
- una interfaz gráfica que reemplace el antiguo menú de texto
- un sistema de carga dinámica de módulos (para así proveer de extensiones en tiempo de ejecución, en vez de en tiempo de compilación)
- portabilidad
- internacionalización (soporte para conjuntos de caracteres no ASCII)

- gestión de memoria real
- arquitectura modular
- instalación multiplataforma
- modo de rescate para aquellos sistemas que no arrancan
- correcciones de antiguos defectos de diseño

El resultado final es una versión de GRUB con una arquitectura más simple, modular y con soporte para caracteres no latinos en el menú de Inicio. A esto hay que añadir un interfaz gráfico para la pantalla de inicio, que puede ser configurado fácilmente mediante un sistema de temas.

En la versión anterior de GRUB, si se quería añadir una entrada al menú de

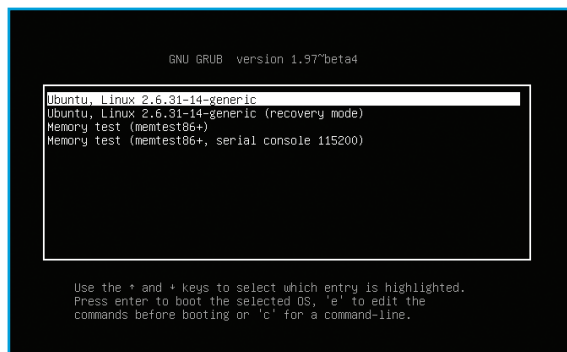


Figura 1: Karmic Koala fue la primera distribución en incluir la nueva versión de GRUB.

GRUB 2 en Ubuntu

Las primeras versiones de GRUB 2 que se lanzaron a la comunidad estaban plagadas de problemas: Ubuntu, por ejemplo, proporcionaba la versión 1.97 beta 4 de GRUB incluida con la actual versión 9.10 de Ubuntu "Karmic Koala" (Figura 1) [3], y fueron muchos los usuarios que comunicaron problemas, incluidos los sistemas que no arrancaban. No obstante, la wiki de Ubuntu tiene soluciones para muchos de ellos. Uno de los problemas más graves fue la protección de la contraseña para el gestor de arranque: Un fallo permitía a los usuarios desbloquear el sistema simplemente adivinando la primera letra de la contraseña [4]. Los desarrolladores corrigieron este error y publicaron una nueva versión. Teniendo en cuenta esta vulnerabilidad crítica, siempre es buena idea estar atento a las nuevas actualizaciones de seguridad. La distribución final de Karmic Koala proporciona una migración completa a GRUB 2. El nuevo GRUB causó problemas en algunas máquinas, especialmente para los usuarios que actualizaron desde Ubuntu 9.04 [5]. Si deseas instalar GRUB 2 en una versión

anterior de Ubuntu (o en otras distribuciones con un sistema de paquetes basados en apt), lanza un terminal y escribe el siguiente comando:

```
$ sudo apt-get install grub2
```

A continuación instala el paquete *grub-pc*, el cual tiene una dependencia con *grub-common*. Con ello tendrás todos los archivos necesarios en su lugar en el equipo. Mientras se instala, se muestra un cuadro de diálogo informándote de que ya tenías instalada GRUB Legacy (la versión antigua) en tu máquina. Confírmalo presionando *OK*, y en el diálogo que sigue, pulsa *Yes* para agregar una entrada al menú actual que te permita ir al GRUB 2 y probarlo.

Después de esto, el proceso de configuración identificará todas tus opciones de inicio en la actual configuración de GRUB y te lo mostrará para su edición. Esto completará la configuración. Después, ambos gestores estarán instalados a la vez en el sistema, aunque sólo se verá la versión antigua del menú que tendrá una entrada extra para el arranque de GRUB 2. Selecciona esta entrada y luego lanza todos tus sis-

temas, uno tras otro, desde el menú de GRUB 2. Si todo funciona como está previsto, lanza un terminal y ejecuta el siguiente comando:

```
$ sudo upgrade-from-grub-legacy
```

El sistema te pedirá que selecciones el dispositivo desde el cual se arrancará el equipo. Asegúrate de seleccionar el dispositivo correcto pulsando la tecla *Espacio*. Si te equivocas, la próxima vez que reinicies el sistema te aparecerá *Error 15* (dispositivo de arranque no encontrado). Si tu sistema te muestra un mensaje de *Error 15*, arranca desde un Live CD, monta la partición raíz en */mnt*, utiliza los siguientes comandos para montar el sistema de ficheros *dev* y luego cambia a un sistema *chroot*:

```
$ sudo mount -bind /dev /mnt/dev
```

```
$ sudo chroot /mnt
```

A continuación ejecuta el comando *dpkg-reconfigure grub-pc* para acceder al proceso de configuración del GRUB 2. Después de seleccionar la entrada adecuada, sal del entorno *chroot* presionando *Ctrl + D*, desmonta el sistema de archivos y reinicia.

inicio o editar una entrada ya existente, tenías que cargar el fichero */boot/grub/menu.lst* en un editor y actualizar el archivo. Con GRUB 2 esto ya no es necesario.

Aunque, si bien es verdad que aún puede encontrarse un archivo *grub.cfg* en el directorio */boot/grub*, éste sólo contiene la configuración que se creó tras la ejecución del script *update-grub*. Este fichero no tiene como objetivo ser modificado por el usuario final.

GRUB 2 automatiza muchos de los pasos que hay que dar para configurar el proceso de carga. Proporciona un conjunto de scripts que nos ayudan a buscar particiones que contienen sistemas operativos conocidos dentro del disco duro de nuestro ordenador y crea las entradas necesarias sin intervención del usuario. Esto es ideal para aquellos sistemas de

arranque dual en el que Linux comparte un mismo ordenador con otro sistema operativo.

La nueva versión también cambia la estructura de ficheros y directorios (ver Tabla 1). En las distribuciones Ubuntu/Debian, el fichero */etc/grub/default* contiene la configuración por defecto de GRUB. Por ejemplo, si no queremos tener que indicar las particiones de arranque utilizando el sistema **UUID**, deberemos escribir en el fichero de configuración la siguiente línea: *GRUB_DISABLE_LINUX_UUID=true*.

Esto nos permitirá poder seguir trabajando con los nombres habituales de los dispositivos.

La mayoría de las opciones no son de utilidad para los usuarios. Durante el proceso de arranque no se muestra la pantalla del menú, pero afortunadamente los desarrolladores nos permiten acceder a esta pantalla si pulsamos la tecla *Mayúsculas* durante la carga. Así

podremos modificar de forma manual la configuración de GRUB en caso de error.

Añadir una Entrada de Menú

Cada vez que instalamos Linux en un sistema, los programas de instalación se encargan de añadir las líneas correspondientes en el menú de arranque, sin embargo, de vez en cuando el proceso se descontrola. Algunos de los errores más habituales que suceden son fallos en el sistema de detección automática o falta de algún parámetro del kernel para la carga del sistema. Para poder realizar un cambio en la configuración de GRUB 2 hay que comprender primero el funcionamiento de éste. Cuando el gestor de arranque GRUB se inicia, arranca todos los scripts alojados en */etc/grub.d/* (¡Acuérdate de ejecutar *sudo chmod u+x nombre_de_script* para establecer los permisos adecuados!).

Los números que aparecen al principio del nombre del archivo garantizan el

Glosario

UUID: Identificador Único Universal, es un estándar para identificadores. Un UUID se compone de un número de 16 bytes, generalmente agrupados en 5 números hexadecimales. Debido al enorme rango de números que puede generar, la probabilidad de duplicar un UUID es virtualmente cero, esto elimina la necesidad de un sistema centralizado de asignación de códigos.

Tabla 1: Archivos y Directorios

<i>/boot/grub/grub.cfg</i>	Fichero de configuración principal de GRUB 2. Se crea de forma automática.
<i>/etc/grub/default</i>	Opciones generales (sin entradas de menú).
<i>/etc/grub.d</i>	Directorio con los scripts auxiliares para la identificación de particiones y de sistemas operativos alojados en ellas. Se pueden crear secuencias de comandos adicionales para agregar su propio menú de entradas según sea necesario.

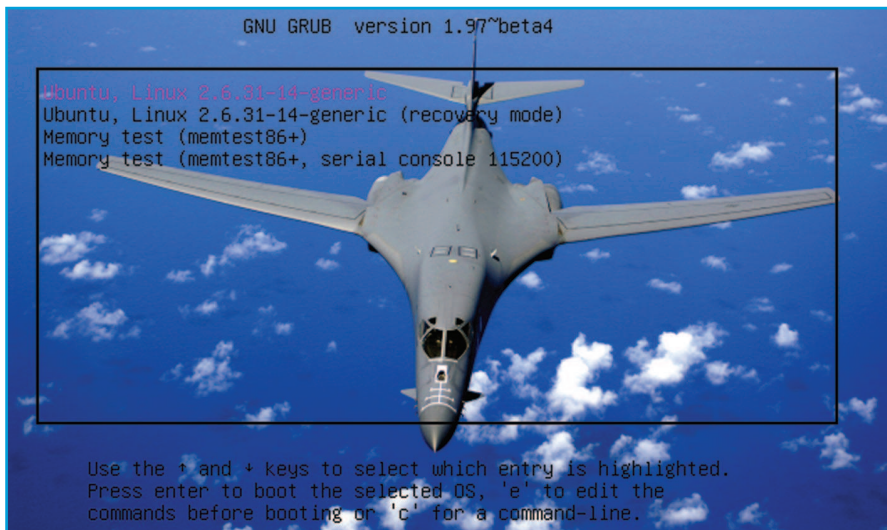


Figura 2: En pocos pasos podemos añadir una imagen de fondo para el menú de arranque.

orden correcto de ejecución. El listado 1 es un ejemplo de un script de inicio personalizado que carga el CD de rescate del sistema desde el disco duro. Nótese cómo el script hace uso de la técnica *here document* para redireccionar la información sobre las líneas de entrada del menú a la salida estándar.

La línea 5 del Listado 1,

```
set root=(hdo,10)
```

indica que queremos que la décima partición del primer disco duro sea usada como sistema de archivos raíz. En esta nueva versión, el esquema de numera-

ción de particiones comienza en 1, en lugar de 0, tal y como lo hacía la versión anterior. Los comandos *linux* e *initrd* establecen la ruta al kernel y al disco RAM de inicio respectivamente. Se pasan dos parámetros al kernel: *subdir* y *setkmap*. La versión anterior de GRUB utilizaba el parámetro *root* para esta tarea. Pero en la nueva se ha convertido en una variable. La lista completa de los comandos GRUB 2 puede consultarse en la página GrubWikiCommandList [6].

Mejoras Visuales

Al arrancar una máquina se nos muestra un menú de inicio de aspecto sobrio y en

dos tonos de color. GRUB 2, sin embargo, puede utilizar como fondo para el menú de arranque imágenes con formato PNG y TGA. Para poder hacerlo en Ubuntu, sólo hay que copiar las imágenes al directorio `/usr/share/images/grub/`. Se pueden reutilizar imágenes ya preparadas instalando el paquete de imágenes: `sudo apt-get install grub2-splashimages`.

Ya sólo nos queda indicar al sistema qué imagen queremos utilizar. Para ello deberemos modificar un archivo del directorio `/etc/grub.d/`. En sistemas Ubuntu/Debian, el fichero que hay que cambiar es `05_debian_theme` (Listado 2). Modificamos la línea 16 para establecer la ruta y el nombre del fichero de la imagen que queremos utilizar y ejecutamos `update-grub` para aplicar los cambios (Figura 2).

Conclusiones

GRUB 2 nos presenta múltiples correcciones y nuevas características, aunque muchos de los cambios no tendrán un efecto en el usuario medio. Una pantalla de inicio con gráficos llamativos es interesante desde un punto de vista de marketing y embellece el escritorio en una fase temprana del proceso de arranque del sistema operativo, pero en realidad no cambia la forma en la que el usuario final interactúa con el sistema. Quizás algo que nos cuesta entender es el motivo por el que se deben utilizar scripts personalizados para ajustar las entradas del menú de inicio.

Por otro lado, el desarrollo de GRUB 2 todavía está en proceso. Las estructuras de control y una estructura modular dan a la configuración mayor flexibilidad, y algún día harán más fácil ampliar el software.

Listado 1: Script de Inicio Personalizado

```
01 echo "Entrada modificada que desde disco duro" {
   lanza un CD de rescate como 05 set root=(hd0,10)
   imagen" >&2                 06 linux /sysrcd/rescuecd
02 cat << EOF                  subdir=sysrcd setkmap=us
03                             07 initrd /sysrcd/initram.igz
04 menuentry "Arranque de      08 }
   Sistema de CD de Rescate    09 EOF
```

Listado 2: Especificación de una Imagen

```
01 #!/bin/bash -e              11 }
02                             12
03 source                       13 # comprobar fondos
   /usr/lib/grub/grub-mkconfig_  utilizables
   ib                           14 use_bg=false
04                             15 if [ "$GRUB_TERMINAL_OUTPUT"
05 set_mono_theme()            ="gfxterm" ]; then
06 {                             16 for i in
07 cat << EOF                   {/boot/grub,/usr/share/images
08 set                           /desktop-base}/moreblue-orbit
   menu_color_normal=white/black -grub.{png,tga} ; do
09 set                           17 if
   menu_color_highlight=black/wh is_path_readable_by_grub $i ;
   ite                           then
10 EOF                           18 bg=$i
```

RECURSOS

- [1] Proyecto PUPA : <http://www.nongnu.org/pupa/>
- [2] Proyecto GRUB 2 : <http://www.gnu.org/software/grub/grub-2.en.html>
- [3] GRUB 2 en Karmic Koala: <https://wiki.ubuntu.com/Grub2>
- [4] Error en contraseña: <https://bugs.launchpad.net/ubuntu/+source/grub2/+bug/478806/>
- [5] Error en actualización: <https://bugs.launchpad.net/ubuntu/+source/grub2/+bug/386789>
- [6] Comandos GRUB : <http://grub.enbug.org/CommandList>