

TRABAJO PRÁCTICO ADM. GNU/LINUX

LABORATORIO GUGLER

INTEGRANTES: Capdevila Juan, Carena Joaquín

TEMA: GRUB (configuración y actualización)

ÍNDICE

PÁGINA 2-3 → *“Introducción y configuración general”*.

PÁGINA 3 → *“Fin configuración general”*.

PÁGINA 3-4 → *“Configuración de arranque por defecto”*

PÁGINA 4-7 → *“Actualización del GRUB”*

PÁGINA 8 → *“Conclusion”*

Copyright (C) 2022, Carena Joaquín, Capdevila Juan.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled "GNU
Free Documentation License"

GRUB

INTRODUCCIÓN

¿Qué es GRUB?

GNU/GRUB (abreviatura de GNU Grand Unified Bootloader) o GRUB es un cargador de arranque, que actualmente es el más usado. Permite editar los parámetros para iniciar un sistema operativo cuando este se encuentra corriendo, es decir, nos da la opción de arrancar uno de los múltiples sistemas operativos que podemos tener en nuestro equipo.

Respecto a su instalación, este se encuentra instalado en el MBR y en parte en un dispositivo de almacenamiento.

Para configurar este cargador de arranque, se usa el archivo de configuración ubicado en `/etc/default/grub` donde se encuentra la lista de sistemas operativos a ejecutar.

En este trabajo nos vamos a encargar de mostrar como configurar GRUB y cómo realizar su respectiva actualización, además de eso vamos a sumar una herramienta que nos puede facilitar las cosas a la hora de usar el GRUB.

CONFIGURACIÓN

A la hora de modificar GRUB hay que tener extremo cuidado ya que es tan delicado que cualquier error puede provocar graves problemas, como por ejemplo, no poder iniciar sesión con ningún sistema operativo.

Por ese motivo, en esta sección nos encargamos de explicarte la manera correcta de configurar GNU/Grub.

Hablemos del archivo de configuración ubicado en `etc/default/grub`. Este contiene:

- * *La partición de arranque por defecto.*
- * *Tiempo de espera (timeout) antes de arrancar automáticamente la partición por defecto.*
- * *Imagen de bienvenida (opcionalmente).*
- * *Particiones de arranque que el usuario puede elegir.*

Mientras que en el contenido del archivo `/boot/grub/grub.cfg` (configuración entera

del gestor de arranque) destacamos algunas líneas importantes:

* *GRUB_DEFAULT=0* → Por defecto, boot se encuentra en la primera partición listada. Por lo que Grub comienza su cuenta desde 0.

* *GRUB_TIMEOUT=10* → Tiempo de espera (segundos) antes de arrancar un sistema operativo automáticamente (el por defecto).

* *GRUB_BACKGROUND=<urlImagen>* → Imagen de fondo en el menú de Grub mientras el usuario selecciona un sistema operativo.

La entrada generalmente por defecto para las distribuciones de Linux es:

```
menuentry 'Debian GNU/Linux' --class debian --class gnu-linux --class gnu --class os $me
nuentry_id_option 'gnulinux-simple-3a92b260-6d21-4345-b218-aa2218a8e710' {
load_video
insmod gzio
if [ x$grub_platform = xxen ]; then insmod xzio; insmod lzopio; fi
insmod part_gpt
insmod ext2
set root='hd0,gpt7'
if [ x$feature_platform_search_hint = xy ]; then
search --no-floppy --fs-uuid --set=root --hint-bios=hd0,gpt7 --hint-efi=hd0,gpt7 --hint-
baremetal=ahci0,gpt7 3a92b260-6d21-4345-b218-aa2218a8e710
else
search --no-floppy --fs-uuid --set=root 3a92b260-6d21-4345-b218-aa2218a8e710
fi
echo 'Cargando Linux 4.2.0-1-amd64...'
linux /boot/vmlinuz-4.2.0-1-amd64 root=UUID=3a92b260-6d21-4345-b218-aa2218a8e710 ro quie
t
echo 'Cargando imagen de memoria inicial...'
initrd /boot/initrd.img-4.2.0-1-amd64
}
```

Donde:

* *menuentry 'Debian GNU/Linux'* → Indica una entrada dentro del Grub que se llamará Debian GNU/Linux.

* *insmod* → Carga un módulo en el kernel de forma permanente.

* *set root='hd0,gpt7'* → Indica la ubicación del sistema operativo. En este caso es en el hd0 (disco duro 0), en la partición 7 (sda7).

* *linux /boot/vmlinuz-4.2.0-1-amd64 root=UUID=3a92b260-6d21-4345-b218-aa2218a8e710 ro quiet* → "linux" indica la imagen del kernel a cargar. "ro" especifica read-only y "quiet" es usado para suprimir los mensajes de arranque descriptivos.

* *initrd /boot/initrd.img-4.2.0-1-amd64* → Especifica la localización del initial ramdisk.

Existen dos formas comunes de configurar Grub. La más simple es realizar un update-grub2 o update-grub y actualizará todo. Las modificaciones más comunes del archivo de configuración ubicado en /boot/grub/grub.cfg (por ejemplo, agregar parámetros de línea de comando al kernel) se realizan en las variables ubicadas en /etc/default/grub.

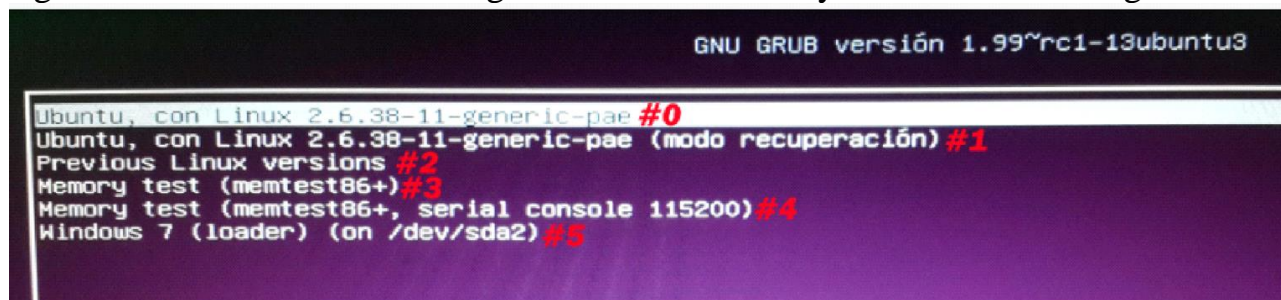
Configurando orden de arranque por defecto con Grub

En esta sección mostraremos paso a paso que realizar para configurar manualmente GNU/Grub de manera exitosa.

1) Abrimos la terminal y ejecutamos el siguiente comando:

gksu gedit /etc/default/grub

2) Editamos el archivo de configuración cambiando el número del parámetro "GRUB_DEFAULT=0" por el que corresponde a nuestro nuevo orden de arranque por defecto, en este caso dejaremos el número 0 ya que queremos que por defecto siga Linux. Una vez hecho esto guardamos el archivo y cerramos el editor gedit.



3) Una vez que cerremos el editor gedit se libera la línea de comandos para ingresar uno nuevo, en ese momento, debemos actualizar Grub con la nueva configuración ejecutando el siguiente comando:

sudo update-grub

4) Reiniciar nuestro equipo y listo, Grub tendrá seleccionado por defecto el nuevo orden de arranque dado anteriormente.

Fuente: <https://www.factor.mx/portal/base-de-conocimiento/configuracion-grub/>

ACTUALIZACIÓN DEL GRUB

Realizar una actualización de GRUB es bastante sencilla pero puede ser compleja para personas que hace muy poco usan Linux, acá realizaremos una actualización del GRUB el cual es de mucha importancia para el proceso de arranque y nos brinda la posibilidad de tener más de un SO en nuestra máquina. Para este ejemplo se usará la distribución Ubuntu pero los pasos son algo similares para las demás distribuciones.

Lo primero que debemos hacer es acceder a la terminal de nuestra distribución, una vez ahí pondremos el siguiente comando:

\$ sudo update-grub

Si el comando funcionó nos debería mostrar por pantalla lo siguiente:

(email protected):~\$ sudo update-grub

(sudo) password for abhishek:

Sourcing file `/etc/default/grub'

Generating grub configuration file ...

Found linux image: /boot/vmlinuz-5.0.0-37-generic

Found initrd image: /boot/initrd.img-5.0.0-37-generic

Found linux image: /boot/vmlinuz-5.0.0-36-generic

Found initrd image: /boot/initrd.img-5.0.0-36-generic

Found linux image: /boot/vmlinuz-5.0.0-31-generic

Found initrd image: /boot/initrd.img-5.0.0-31-generic

Found Ubuntu 19.10 (19.10) on /dev/sda4

Found MX 19 patito feo (19) on /dev/sdb1

**Adding boot menu entry for EFI firmware configuration
done**

¿Qué pasa si no se encuentra la orden de update-grub?

Existe la posibilidad de que en nuestra distribución no se encuentre la orden de update-grub disponible.

En este escenario posible no hay que alarmarse ya que la orden de update-grub es simplemente un código auxiliar para la ejecución del siguiente comando “grub-mkconfig -o /boot/grub/grub.cfg” que la función que realiza es la de generar el archivo de configuración de grub2.

Lo que debemos hacer para poder actualizar el GRUB con éxito es insertar la siguiente línea de comando en la terminal:

\$ sudo grub-mkconfig -o /boot/grub/grub.cfg

*Observación: este comando sirve para cualquier distribución de Linux que no funcione con la orden de update-grub

¿Es realmente necesario actualizar el menú GRUB en tu sistema?

Podría decirse que hay muchas situaciones en las que necesitarás tener actualizado el menú grub.

Un caso sería suponer que deseo cambiar el grub archivo de configuración (/etc/default/grub) para poder cambiar el orden de arranque o reducir el tiempo de arranque por defecto. Todos estos cambios no los puedo realizar a menos que tenga el grub actualizado.

Otro escenario teórico sería el tener más de una distribución Linux en el mismo sistema y deseas que aparezcan en el gestor de arranque.

Un ejemplo de esto sería. Tengo dos discos o dos particiones, en el primer disco hay un sistema Ubuntu 19.10 y luego de eso instale Ubuntu pero en este caso la versión 18.04 el segundo Sistema Operativo (18.04) también instalado con su respectivo GRUB y ahora la pantalla de GRUB es controlada por el GRUB de Ubuntu 18.04. En el segundo disco, he instalado el MX Linux, pero yo no instalé GRUB en el momento de la instalación. Quiero que el GRUB existente (controlado por Ubuntu 18.04) pueda manejar todas las entradas de sistema operativo que posea mi máquina.

Este ejemplo no es muy usual pero puede llegar a suceder, Siguiendo con el ejemplo para poder visualizar el MX Linux que instale, necesitamos realizar una actualización al GRUB de Ubuntu 18.04.

Simplemente con realizar esta actualización el GRUB podrá encontrar las distintas versiones de Kernel de los sistemas instalados y el nuevo MX linux de la otra partición o disco.

Una herramienta de gran utilidad para controlar nuestro GRUB

Una herramienta muy útil para poder administrar nuestro gestor de arranque GRUB es la de *grub customizer* es muy cómodo tanto en apariencia como en configuración dejando atrás los engorrosos comandos que gente novata en el campo no entiende. La instalación en la distribución Ubuntu es de la siguiente manera.

Primero agregaremos los repositorios en nuestro sistema con el siguiente comando:

```
$ sudo add-apt-repository ppa:danielrichter2007/grub-customize
```

Ahora actualizaremos la lista de repositorios con el siguiente comando:

```
$ sudo apt-get update
```

Y finalmente instalaremos la respectiva herramienta con el siguiente comando:

```
$ sudo apt-get install grub-customizer
```

Una imagen ilustrativa de la instalación:

```
fish /home/gonka79
Archivo Editar Ver Buscar Terminal Ayuda
Ign:41 ftp://ftp.mondorescue.org//debian 8 Release.gpg
Leyendo lista de paquetes... Hecho
W: Error de GPG: http://download.opensuse.org/repositories/home:/stevenpusser/xUbuntu_18.04 InRelease: Las firmas siguientes no se pudieron verificar porque su clave pública no está disponible: NO_PUBKEY 0FAD31CA8719FCE4
E: El repositorio «http://download.opensuse.org/repositories/home:/stevenpusser/xUbuntu_18.04 InRelease» no está firmado.
N: No se puede actualizar de un repositorio como este de forma segura y por tanto está deshabilitado por omisión.
N: Vea la página de manual apt-secure(8) para los detalles sobre la creación de repositorios y la configuración de usuarios.
W: Error de GPG: ftp://ftp.mondorescue.org//debian 8 Release: Las firmas siguientes no se pudieron verificar porque su clave pública no está disponible: NO_PUBKEY 6BA8C2D220EBFB0E
E: El repositorio «ftp://ftp.mondorescue.org//debian 8 Release» no está firmado.
N: No se puede actualizar de un repositorio como este de forma segura y por tanto está deshabilitado por omisión.
N: Vea la página de manual apt-secure(8) para los detalles sobre la creación de repositorios y la configuración de usuarios.
[gonka79@gonka79-Mint:~] [~]
$ sudo apt-get install grub-customizer
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
baloo-kf5 brave-keyring catdoc dolphin epiphany-browser-data kinit kio
kpackagekit5 libappindicator1 libdbusmenu-gtk4 libdbusmenu-qt5-2
libdolphinvc5 libepub0 libfam0 libindicator7 libkf5archive5 libkf5attica5
libkf5auth-data libkf5auth5 libkf5baloo5 libkf5baloengine5
libkf5baloowidgets-bin libkf5baloowidgets-data libkf5baloowidgets5
libkf5bookmarks-data libkf5bookmarks5 libkf5codecs-data libkf5codecs5
libkf5completion-data libkf5completion5 libkf5config-data libkf5configcore5
libkf5configgui5 libkf5configwidgets-data libkf5configwidgets5
libkf5coreaddons-data libkf5coreaddons5 libkf5crash5 libkf5dbusaddons-data
libkf5dbusaddons5 libkf5declarative-data libkf5declarative5 libkf5doctools5
libkf5filemetadata-bin libkf5filemetadata-data libkf5filemetadata3
libkf5globalaccel-data libkf5globalaccel5 libkf5guiaddons5 libkf5i18n-data
libkf5i18n5 libkf5iconthemes-data libkf5iconthemes5 libkf5idle5
libkf5itemviews-data libkf5itemviews5 libkf5jobwidgets-data
libkf5jobwidgets5 libkf5js5 libkf5kcmutils-data libkf5kcmutils5
libkf5kdelibs4support-data libkf5kdelibs4support5 libkf5khtml-data
libkf5khtml5 libkf5kiocore5 libkf5kiofilewidgets5 libkf5kiogui5
libkf5kiontlm5 libkf5kiowidgets5 libkf5konq-data libkf5konq6
```

Fuente: <https://electronicssoftware.net/como-actualizar-grub-en-ubuntu-y-otros-linux/>

CONCLUSION

En conclusión a todo lo que se vio en este trabajo de investigación es que el GRUB es de gran importancia, sobre todo para las personas que se quieren iniciar en el mundo de Linux pero todavía no están del todo seguros de dar ese cambio de S.O. Ya sea por la comodidad que le brinda su sistema o por otro hecho cualquiera.

El GRUB nos ayuda a poder tener el sistema que más nos guste y que mejor acostumbrados estemos sin cerrarnos la posibilidad de usar otros. Ya que con la función del GRUB podremos ir intercalando entre ambos o más Sistemas Operativos que tengamos instalados en nuestra máquina.

Puede ser difícil de aprender a usarlo en un principio pero con tiempo y paciencia se logra tener un uso fluido del mismo. Además existen programas que nos sirven como herramientas del GRUB y nos otorgan un uso más cómodo y vistoso.