

Instalando gentoo, por Víctor Orozco

Preliminares

Lo que la mayoría de personas se pregunta, es ¿porqué elaborar un nuevo manual cuando todo esta dicho en el handbook?. Por una o varias razones esta guía no pretende ser ni mejor ni ser reemplazo del handbook, de hecho esta basado en el mismo. Esta guía no esta diseñada para aquellos que ya tengan el valor de empezar con instalaciones avanzadas, va dirigida a aquellas personas a las cuales el handbook les parece largo y tedioso. Además que no tienen los suficientes conocimientos técnicos pero de una u otra manera necesitan Gentoo instalado. Como un extra incluyo algunos links utiles que me han servido a lo largo de mi experiencia (poca a decir verdad) con Gentoo. Este manual esta pensado para arquitecturas i686.

Paso 1 - Consiguiendo todo lo necesario.

Para instalar gentoo necesitamos únicamente un CD minimal para arrancar el sistema base con el cual realizaremos la instalaciones.

Después de arrancar tendremos un sistema operativo básico corriendo sobre memoria RAM, al inicio se nos mostrara un prompt con privilegios de superusuario, similar a este:

```
livecd root #
```

Es momento de usar los dedos y dejar el mouse por un rato.

Notese que una instalación normal es totalmente vía web, si la red no tuviera un servidor DHCP (como el que se encuentra en la mayoría de routers y modems ADSL caseros) el comando net-setup nos presenta un asistente básico para configurar nuestra conexión a internet. Para verificar que nuestra dirección ip sea la correcta, podemos ejecutar el comando ifconfig.

```
livecd root # ifconfig eth0
eth0  Link encap:Ethernet  HWaddr 00:A0:D1:4F:37:17
      inet addr:10.10.4.196  Bcast:10.10.4.255  Mask:255.255.255.0
      UP BROADCAST MULTICAST  MTU:1500  Metric:1
      RX packets:0 errors:0 dropped:0 overruns:0 frame:0
      TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
      Interrupt:17 Base address:0xa000
```



Paso 2 - Preparando nuestro disco duro

Es momento de preparar nuestro disco duro, para esto la herramienta predilecta es fdisk, sin embargo cuando no tenemos conocimientos de las características de lo que estamos haciendo podemos utilizar cfdisk que seria la versión amigable.

```
livecd root # cfdisk
```

```
File Edit View Terminal Tabs Help
cfdisk (util-linux-ng 2.13.1.1)

Disk Drive: /dev/sda
Size: 60011642880 bytes, 60.0 GB
Heads: 255 Sectors per Track: 63 Cylinders: 7296

-----
Name      Flags      Part Type  FS Type      [Label]      Size (MB)
-----
sda1      Boot      Primary    NTFS          [^A]          10487.24
sda2                        Primary     Linux ext3    32637.92
sda3                        Primary     Linux swap / Solaris 1151.54
sda5      Logical   Linux ext3   [/home]       15726.74
                        Logical     Free Space    8.23

[Bootable] [ Delete ] [ Help ] [Maximize] [ Print ]
[ Quit ]   [ Type ]  [ Units ] [ Write ]

Toggle bootable flag of the current partition
```

El esquema de particiones puede ser personal, recomiendo uno similar a este:

Partición	Punto de montaje	Tamaño	Sistema de fichero
/dev/hda1	/boot	100 MB	ext2
/dev/hda2	none	Doble de la memoria RAM	swap
/dev/hda3	/	15000 MB	reiserFS/ext3
/dev/hda4	/home	resto de espacio libre	reiserFS/ext3

Dependiendo si escogemos o no el esquema anterior nos queda dar formato a cada una de las particiones con mke2fs para ext2/ext3 y mkreiserfs si vamos a usar reiser. Se indica si sera ext2 o ext3 agregando la opción -j para activar journal. También en este punto podemos activar nuestra partición swap.

```
# mke2fs -j dev/hda3
# mke2fs -j /dev/hda4
# mke2fs /dev/hda1
# mkswap /dev/hda2
# swapon /dev/hda2
```



Paso 3 - configuramos el sistema

Hechas nuestras particiones nos queda montarlas para poder acceder a ellas, regularmente el punto de montaje sera /mnt/gentoo, pero puede ser cualquier otro dentro de la carpeta mnt siempre y cuando creamos la carpeta de destino.

```
# mount /dev/hda3 /mnt/gentoo
# mkdir /mnt/gentoo/boot
# mkdir /mnt/gentoo/home
# mount /dev/hda4 /mnt/gentoo/home
# mount /dev/hda1 /mnt/gentoo/boot
```

En este punto es conveniente configurar la fecha y hora de nuestro sistema mediante el comando date, el formato es el siguiente date MMDDhhmmYYYY.

```
# date 080800002008
```

Paso 4 - Descargando el Stage y Portage

Stage 1,2 o 3, sera nuestro sistema base para ir construyendo nuestro sistema a antojo. La regla es a mayor numero de stage menor optimización, aunque el único oficialmente soportado por gentoo hasta la fecha es el stage 3.

El arbol de portage, son el corazón y alma de gentoo. Son una serie de scripts llamados ebuilds que contienen las dependencias, opciones e instrucciones de compilación de los paquetes oficialmente soportados por gentoo.

Recordando que nuestros discos duros están sobre /mnt/gentoo procedemos a descargar y descomprimir stage y portage respectivamente, podemos navegar por los mirrors de descarga de gentoo con el navegador en modo texto links, en la url <http://www.gentoo.org/main/en/mirrors.xml>.

Suponiendo que queramos un mirror de Norteamérica, utilizamos wget para descargarlo. Notese que usamos la variante i686, que aplica a cualquier procesador mas halla de los pentium originales de intel (recomendado para pentium III en adelante).

Primero que nada entramos a lo que se convertira en nuestra instalación de gentoo o de lo contrario vamos a descargar el stage en nuestra memoria ram

```
# cd /mnt/gentoo
```

Luego descargamos el stage

```
# wget ftp://gentoo.arcticnetwork.ca/pub/gentoo/releases/x86/2008.0/stages/stage3-i686-2008.0.tar.bz2
# wget ftp://gentoo.arcticnetwork.ca/pub/gentoo/snapshots/portage-latest.tar.bz2
```

OPCIONAL: Gentoo tiene un nuevo proyecto para producir stage más actualizados de forma semanal, al dia de hoy utilizare el stage del 04 de agosto de 2009 como un ejemplo por si quisieramos utilizar este stage, en todos los mirrors se encuentra dentro de "releases/arquitectura/autobuilds"

```
# wget
ftp://gentoo.arcticnetwork.ca/pub/gentoo/releases/x86/autobuilds/20090804/stage3-i686-20090804.tar.bz2
```

Ya que tenemos el stage procedemos a desempaquetar lo ya descargado con el comando tar, las opciones significan

- x – Extract
- v – Verbose
- j – Sistema de archivos bz2
- p – Preservar sistema de archivos y symlinks
- f – Utilizar el sistema de archivos predeterminado

```
# tar xvjpf stage3-i686-2008.0.tar.bz2
```

O si bajamos el más reciente (como el ejemplo)

```
# tar xvjpf stage3-i686-20090804.tar.bz2
```

Y por ultimo descomprimos portage

```
# tar xvjfb /mnt/gentoo/portage-latest.tar.bz2 -C /mnt/gentoo/usr
```

Paso 5 - Seleccionando nuestros orígenes de descarga

Gentoo descargara el código fuente desde internet, aunque es código fuente gentoo tiene mirrors diseñados para esto, aunque este paso es opcional es bueno que escojamos desde donde gentoo debe descargar los paquetes. Mediante mirror select podemos elegir el(los) mirror(s) desde donde gentoo descargara, y mediante una tubería >> vaciarlo en nuestro archivo make.conf que explicaremos mas adelante.

```
# mirrorselect -i -o >> /mnt/gentoo/etc/make.conf
```



También podemos elegir los mirrors rsync, portage tiene la particularidad de actualizarse en diferido, gracias al protocolo rsync solo descarga los ebuild que no esten actualmente en nuestro sistema y elimina los obsoletos.

```
# mirrorselect -i -r -o >> /mnt/gentoo/etc/make.conf
```

Hasta este punto, hemos llegado antes de cambianos a nuestro nuevo sistema gentoo, como consejo adicional deberíamos de copiar la información de nuestro servidor DNS para proseguir con la instalación

```
# cp -L /etc/resolv.conf /mnt/gentoo/etc/resolv.conf
```

Paso 6 – Llego la hora del chrooting

La instrucción chroot nos permitirá ejecutar comandos o shells interactivos con un directorio raíz especial. En este caso nuestro nuevo sistema que actualmente esta en /mnt/gentoo, ademas de esto debemos de montar el sistema de archivos proc (que representa algunos procesos del sistema y su información) y dev (que representa los dispositivos de nuestro sistema), esto ayudara posteriormente cuando configuremos el kernel linux a utilizar

```
# mount -t proc none /mnt/gentoo/proc
# mount -o bind /dev/ /mnt/gentoo/dev
```

Bueno no queda mas que hacer chroot utilizaremos bash como interprete de comandos

```
# chroot /mnt/gentoo /bin/bash
```

Actualizamos nuestras variables de entorno

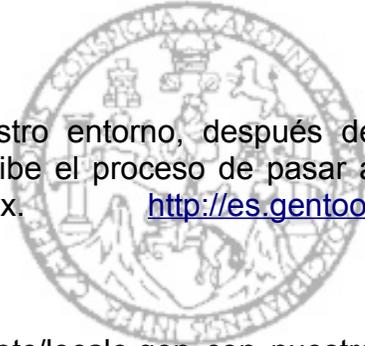
```
# env-update && source /etc/profile
```

Y podemos poner la leyenda (chroot) en nuestro interprete de comandos para facilidad, aunque puede ser cualquier otra leyenda.

```
# export PS1="(chroot) $PS1"
```

Si por una u otra razón la instalación la hemos dejado pendiente uno o dos días, podemos en este momento actualizar portage, no es recomendable actualizar más de una vez en 24 horas, ya que si hacemos esto gentoo añadirá nuestra ip a una lista negra o ban list.

```
# emerge -sync
```



Paso 7 – Mas configuraciones

Para proseguir con la instalación es necesario configurar nuestro entorno, después de hechas estas configuraciones recomiendo el siguiente link, describe el proceso de pasar a idioma español nuestro gentoo linux. http://es.gentoo-wiki.com/HOWTO_Castellanizar_Gentoo

La configuración básica seria la siguiente, editamos el archivo `/etc/locale.gen` con nuestro editor de texto favorito (en mi caso nano).

```
# nano -w /etc/locale.gen
```

Un ejemplo de archivo seria el siguiente:

```
en_US ISO-8859-1
en_US.UTF-8 UTF-8
#soporte para el español de España
es_ES@euroISO-8859-15
es_ES@euro UTF-8
#Chapinizando (de Guatemala) un poco gentoo
es_GT ISO-8859-15
es_GT UTF-8
```

Por ultimo activamos los cambios y activamos las variables locales mediante el comando:

```
# locale-gen
```

Ademas de esto hay que configurar nuestra zona horaria, las disponibles están dentro de `/usr/share/zoneinfo`, un `ls` dentro de esta carpeta puede aclararnos el panorama, las que apliquen a nuestro sistema van dentro de `/etc/localtime`, asi que procedemos a copiarlas:

```
# cp /usr/share/zoneinfo/America/Guatemala /etc/localtime
```

Paso 8 – El kernel linux

Es tal vez el paso más fundamental pero a la vez difícil, gentoo en este caso nos proporciona dos caminos, la vía tradicional y mediante la herramienta `genkernel` que genera un kernel similar al del live cd que evaluá nuestros dispositivos y decide que módulos cargar o no, ya que la vía `genkernel` es ta mas tardada (en compilar), para este tutorial la compilación sera manual, recomiendo la vía `genkernel` para las personas que inician en gentoo, porque si no tienen mucho conocimiento de su hardware terminaran con un bonito kernel panic antes de arrancar el sistema.

Algunos links utiles:

Guia genkernel: <http://www.gentoo.org/doc/es/genkernel.xml>

Compilando un kernel manualmente: http://es.gentoo-wiki.com/HOWTO_Compilar_el_kernel_manualmente



Primero obtenemos el código fuente del kernel, portage nos facilita esta tarea (para información acerca de portage visitar: http://es.gentoo-wiki.com/HOWTO_Usar_Portage_Correctamente).

Nuestro kernel puede ser de distintas variantes, entre las que he utilizado están la variante vanilla que es tal y como lo produce kernel.org. Gentoo-sources que es un kernel vanilla con agregados propios de gentoo como el parche fbcondecor. Tuxonice-sources igual que el anterior solo que con el parte extra para hibernacion mediante tuxonice. Para este tutorial utilizaremos una variante gentoo sources.

```
# emerge gentoo-sources
```

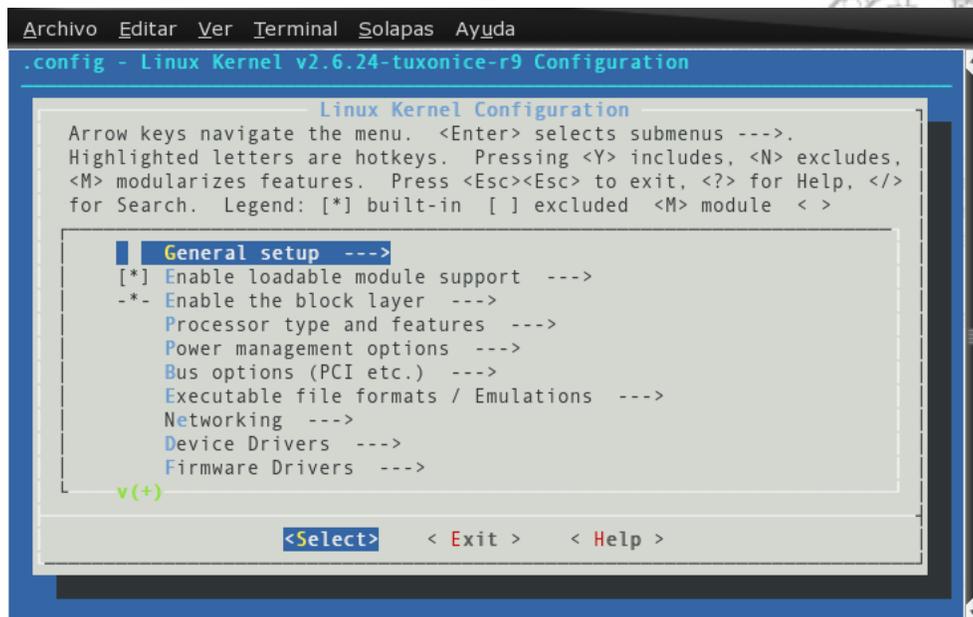
Ahora es momento de hacer un symlink para la carpeta /usr/src/linux, en sistemas operativos GNU/Linux tenemos la versatilidad de poder arrancar con distintas versiones de kernel, este enlace simbolico apuntaria a la version actual, para no profundizar el tema definamos a /usr/src/linux como la carpeta donde drivers de terceros buscaran el código fuente del kernel.

```
# cd /usr/src && ln -s linux-2.6.versionquedesgargamos linux
```

Bueno ahora si a configurar nuestro kernel

```
# cd linux
# make menuconfig
```

El menu que se presenta es una interfaz para editar el archivo .config dentro de la carpeta /usr/src/linux, que define que módulos del kernel se compilaran o no y cuales irán directamente o como módulos externos, no profundizare en el tema ya que si se eligió esta vía significa que tenemos al menos nociones de nuestro hardware y las opciones del kernel, para los que se asusten el ver esta pantalla les recomiendo que mejor vean el link que han dejado arriba, de la guía genkernel, la pantalla sera similar a esta:



Seria mentira decirles que la configuración que me funciono a mi les funcionara. Para esto es más facil que pongan

Luego de configurado solo nos queda compilar el kernel, es momento de hacernos un café ya que dependiendo de cuantos módulos activemos así sera el tiempo.

```
# make && make modules_install
```

Bueno luego de un par de minutos y un café ya solo queda copiar nuestro nuevo kernel hacia la partición boot y el system map, como el objetivo de esta guía no es complicar la existencia definiremos bzImage como la imagen del kernel y el archivo System.map como una tabla de símbolos para el kernel (mas información: <http://dirac.org/linux/system.map/>, en ingles).

```
# cp System.map /boot/System.map-versiondekernel
```

```
# cp arch/i386/boot/bzImage /boot/bzImage-versiondekernel
```

No es necesario que utilicemos esos nombres, incluso se podrían llamar "esteesmikernel". El uso de un nombre estándar nos ayudara en futuras instalaciones de kernel y configuración de nuestro gestor de arranque.



Paso 9 – Configurando fstab

Fstab, es un fichero para definir puntos de montaje (recordemos que en UNIX todo cuelga de una única raíz /) que serán cargados al arrancar el sistema. Cada línea define una partición y lleva mas o menos esta estructura.

- Partición
- Punto de montaje
- Tipo de sistema de archivos
- Opciones especiales
- Determina si se debe volcar o no la partición (si no sabe que es esto solo dejen 0)
- Orden en que fsck debe comprobar en caso de que el sistema no se apague correctamente (si tampoco sabes que es deja todos 0 y boot 1)

```
# nano -w /etc/fstab
```

Retomando nuestro esquema de particiones inicial debería de verse mas o menos asi:

/dev/hda1	/boot	ext2	defaults	1	2
/dev/hda2	/none	swap	sw	0	0
/dev/hda3	/	ext3	defaults	0	1
/dev/hda4	/home	ext3	defaults	0	2
shm	/dev/shm	tmpfs	nodev,nosuid,noexec	0	0
proc	/proc	proc	nodev,nosuid,noexec	0	0

Si tenemos mas dudas dentro de este mismo archivo vienen las especificaciones, asi que no hay que preocuparse.



Paso 10 – Mas configuraciones

Red

Establecemos el nombre que tendrá nuestro equipo en la red:

```
# nano -w /etc/conf.d/hostname
```

Reemplazar "localhost" por cualquier otro nombre.

Si nuestra pc fuera parte de un dominio

```
# nano -w /etc/conf.d/net
```

Y establecemos el dominio: `dns_domain_lo="dominio"`.

Habilitar el demonio dhcp para determinadas interfaces

```
# nano -w /etc/conf.d/net
```

Con estas lineas bastara

```
config_eth0=( "dhcp" )  
dhcp_eth0="nodns nontp nonis"
```

Por ultimo activemos la interfaz para que arranque junto don el sistema

```
# rc-update add net.eth0 default
```

rc-update es el comando para editar los runlevels, y default al nivel que acabamos de añadir net.eth0.

Modificamos el archivo `/etc/rc.conf` que son las configuraciones globales iniciales, el archivo esta bastante bien comentado queda a nuestro criterio que cambios hacerle.

```
# nano /etc/rc.conf
```

IMPORTANTE:

Debemos de cambiar la clave de el usuario root (superusuario) ya que de reiniciar sin este cambio no podremos administrar el sistema con privilegios.

```
# passwd
```

KEYMAPS

Establecemos nuestra variante de teclado

```
# nano -w /etc/conf.d/keymaps
```

Activaremos las siguientes líneas:

```
KEYMAP="es"  
SET_WINDOWKEYS="yes"
```



Configuración del reloj

El reloj se configura de la siguiente manera:

```
# nano -w /etc/conf.d/clock
```

Al igual que el anterior esta bastante bien comentado, algunas líneas que configure son:

```
CLOCK="UTC"  
TIMEZONE="America/Guatemala"  
CLOCK_OPTS=""  
CLOCK_SYSTOHC="yes"
```

Paso 11 – Servicios

En UNIX tenemos la particularidad de tener servicios de sistema, que están corriendo en backend para ser llamados por los programas que los necesiten, algunos de los más importantes que deberíamos instalar:

Registro de sistema (log)

En mi caso utilizo syslog

```
# emerge syslog-ng  
# rc-update add syslog-ng default
```

Servicio cron

Para ejecutar tareas programadas en determinado tiempo (en mi caso vixie-cron)

```
# emerge vixie-cron  
# rc-update add vixie-cron default
```

Indexado de ficheros (para búsquedas rápidas con herramienta locate)

```
# emerge slocate
```

Cliente dhcp (para obtener los datos de la red de manera automática)

```
# emerge net-misc/dhcpd
```

Paso 12 – Gestor de arranque (GRUB)

El gestor de arranque es una pieza fundamental para el correcto funcionamiento del sistema en el definiremos sistemas operativos (o versiones de kernel a utilizar), tenemos dos opciones grub o lilo, en mi caso me decido por grub ya que es el que siempre he utilizado.



Instalamos grub

```
# emerge grub
```

La creación del menu de arranque se hace mediante el archivo grub.conf así que procedemos a editarlo, este archivo es muy personalizable, así que solo definiremos un ejemplo de lo necesario para que nuestro sistema arranque

```
# nano -w /boot/grub/grub.conf
```

En nuestro ejemplo sería algo así:

```
default 0
timeout 30

title=Mi primer gentoo
root (hd0,0)
kernel /boot/bzImage-versiondekernel root=/dev/hda3
```

Por último pero no menos importante instalamos el gestor en el disco duro-

```
# grep -v rootfs /proc/mounts > /etc/mtab
# grub-install --no-floppy /dev/hda
```

Ya casi, por el momento reiniciamos el sistema, y si todo va bien, nos vemos en un par de minutos

Reiniciamos el sistema

```
# reboot
```

Ya de vuelta nos encontramos con gentoo funcionando, iniciamos sesión como root y faltaría un último paso, crear nuestro usuario de día a día, root solo es necesario para administrar el sistema ya que el tiene el poder de editar cualquier archivo es peligroso utilizarlo más allá de administración, por defecto lo agregamos a los grupos de usuario que recomienda el handbook de gentoo y establecemos que el interprete de comandos a utilizar será bash

```
# useradd -m -G users,wheel,audio,cdrom,usb,video -s /bin/bash nombre_de_usuario
```

Hecho esto podemos iniciar sesión con nuestro nuevo usuario.

Y si ya llegaste hasta aqui Felicidades!!! ya tenemos gentoo instalado.

Desde aquí sera cuestión de nosotros instalar paquetes y personalizar el sistema, algunos links utiles:

Portage(en ingles): <http://www.gentoo.org/doc/en/handbook/handbook-x86.xml?part=2&chap=1>

ALSA (sonido): <http://www.gentoo.org/doc/es/alsa-guide.xml>

Gnome (entorno de escritorio): <http://www.gentoo.org/doc/es/gnome-config.xml>

Kde(en ingles): <http://www.gentoo.org/proj/en/desktop/kde/kde-config.xml>

Fluxbox(en ingles): http://gentoo-wiki.com/HOWTO_Fluxbox

