

Introducción a GNU/Linux

Índice

| | |
|---|---|
| Introducción a GNU/Linux..... | 1 |
| 1. ¿Qué es GNU/Linux?..... | 2 |
| 1.1. ¿Qué es un sistema operativo?..... | 2 |
| 1.2. ¿A qué me refiero con "libre"?..... | 2 |
| 2. Distribuciones, ¿qué es eso?..... | 3 |
| 2.1. Debian, otro mundo es posible..... | 3 |
| 2.2. Knoppix, lleva tu GNU/Linux contigo..... | 3 |
| 3. Falsos tópicos acerca de GNU/Linux..... | 4 |
| 4. Ventajas de GNU/Linux..... | 4 |
| Glosario..... | 5 |

1. ¿Qué es GNU/Linux?

GNU/Linux es un sistema operativo libre. de aquí pueden surgir dos preguntas: 1) qué es un sistema operativo y 2) qué quiero decir con libre.

1.1. ¿Qué es un sistema operativo?

Un sistema operativo es el conjunto de programas (por decirlo de una manera sencilla) que dan utilidad al ordenador: sin un sistema operativo, un ordenador no sería más que un puñado de silicio, cobre y hierro.

El sistema operativo consta de un Núcleo (kernel) y un conjunto de utilidades básicas para manejar el sistema. En nuestro caso, el Núcleo del sistema es Linux, un kernel creado por Linus Torvalds en 1991 bajo licencia libre y que hoy día tiene un desarrollo muy activo. El resto de utilidades es lo que se llama el sistema GNU. Estas utilidades fueron creadas por la Free Software Foundation para el Proyecto GNU, un proyecto creado por Richard M Stallman con la intención de diseñar un sistema operativo libre por completo.

1.2. ¿A qué me refiero con "libre"?

Cuando hablo de un programa libre, no me estoy refiriendo exactamente a que sea gratis. Suele ser así la mayor parte de las veces pero las licencias de software libre no implican que éste deba ser gratuito. Lo que nosotros llamamos software libre es aquel que, como su propio nombre indica, deja libertad al usuario. Al ser distribuido junto al Código Fuente, el programa puede ser modificado, a la vez que estudiado para ver como funciona y que hace exactamente. Esto último puede importar poco a mucha gente, ya que piensan que, al no saber programación les da igual tener o no el Código Fuente ya que no lo van a entender. Sin embargo, hay que contar con que hay un inmenso grupo de desarrolladores (programadores) por todo el mundo que han analizado ese software de forma imparcial y se han asegurado que dichos programas cumplen la función que prometen hacer.

Las directrices de Software Libre de Debian definen que el software es libre cuando:

1. " La licencia del programa no debe restringir su posterior venta o distribución. Tampoco debe pedir ninguna clase de *royalties* por ello "

Es decir, una vez ha llegado el programa a tus manos, debes ser capaz de distribuirlo como mejor te parezca sin que el autor tenga que percibir ningún beneficio. Por supuesto, amablemente aceptará tus donaciones.

2. " El programa debe incluir su Código Fuente y debe permitir que sea distribuido bajo los mismos términos que el resto del software "

El Código Fuente de un programa es algo así como su receta. Es lo que los programadores escriben al hacer el programa en lenguajes más o menos parecidos al lenguaje humano y que luego compilan (traducen a un lenguaje que entiende la máquina). El acceso al Código Fuente nos permite comprender como funciona exactamente un programa, encontrar fallos, comprobar si el programa hace lo que dice hacer,...

3. " La licencia debe permitir modificaciones y trabajos derivados, y debe permitir que éstos se distribuyan bajo los mismos términos que la licencia del programa original. "
4. " La licencia puede restringir la distribución del código fuente en forma modificada *_sólo_* si la licencia permite la distribución de *ficheros de parches* junto con el código fuente con la intención

de modificar el programa en el momento de compilarlo. La licencia debe permitir explícitamente la distribución de software a partir de código fuente modificado. La licencia puede obligar a los trabajos derivados a llevar un nombre o número de versión diferentes del programa original. (Esto es un compromiso. El grupo de Debian anima a todos los autores a no restringir ningún fichero, fuente o compilado, de ser modificado.) "

De este modo, a cada usuario le llega exactamente lo que el autor escribe y luego, mediante esos *ficheros de parches* se pueden aplicar las modificaciones que ha hecho el resto de la gente.

5. " La licencia no debe discriminar a ninguna persona o grupo de personas. Tampoco puede restringir a nadie que trabaje en un determinado campo a no usar el programa. Por ejemplo, no puede restringir el uso del programa a empresas, o ser usado en investigación genética. "

2. Distribuciones, ¿qué es eso?

GNU/Linux no es como otros sistemas mas conocidos por muchos de los usuarios normales de ordenadores en lo que a las distribuciones se refiere.

En un SO propietario como Windows® estamos acostumbrados a que Microsoft decida que debemos usar y nos vende todo como un paquete cerrado. Sin embargo, y como he dicho, GNU/Linux es sólo el Nucleo. A éste hay que acompañarle con más programas que den utilidad al sistema. De eso se encargan las distribuciones. Pueden estar mantenidas por empresas (RedHat, Mandrake, SuSe) o bien por una comunidad de desarrolladores (Debian) que son los que se encargan de juntar los distintos programas, compilarlos, empaquetarlos y hacer que funcionen bien entre ellos.

2.1. Debian, otro mundo es posible

Debian es, probablemente, la mejor distribucion de GNU/Linux por:

1. **Es 100% libre.** Debian sigue la filosofia de mantenerse 100% libre y no incluye en sus distribuciones oficiales que no cumplan sus directrices acerca de qué es libre y qué no lo es
2. **No es comercial.** Al estar mantenida por una comunidad de desarrolladores que dedican a Debian su tiempo libre, no está sometida a las presiones que pueda tener una empresa. Esto implica que no salen con mucha frecuencia nuevas versiones pero estas se liberan sin prisas y convenientemente probadas. Sin embargo, aunque no salgan versiones estables tan a menudo como en otras distribuciones, las versiones "inestables" tienen una actualización casi diaria.
3. **El sistema de paquetes es muy bueno.** Una de las diferencias principales entre una distribución y otra es el sistema de paquetes, o como se organiza la instalación y desinstalación de programas. Normalmente instalar un programa es tan sencillo como hacer: **apt-get install nombredelprograma**

2.2. Knoppix, lleva tu GNU/Linux contigo

Knoppix es lo que se suele llamar un *Live-CD*. Esto es un sistema operativo dentro de un cd autoarrancable. Hablando claro: metes el cd, enciendes el ordenador y arranca un bonito sistema GNU/Linux completamente funcional con el que puedes navegar por internet, jugar, escribir charlas como esta...

3. Falsos tópicos acerca de GNU/Linux

Existen falsos mitos acerca de GNU/Linux, bien por desconocimiento, por FUDs (falsos rumores creados por enemigos de otras compañías que sienten pánico ante tal competencia) o también hay muchos casos de gente que probó GNU/Linux hace unos cuantos años, cuando resultaba poco asequible para la mayoría de usuarios de nivel bajo

- **GNU/Linux es difícil de usar.** Como acabo de decir, eso suele decirlo gente que ha probado versiones viejas cuya usabilidad para el usuario novato era mínima. Puede que ahora mismo le resulte complicado aprender GNU/Linux a alguien que lleva usando años un SO tipo Windows. La cuestión no es que GNU/Linux sea difícil o fácil de usar, lo que puede resultar costoso a alguien es el cambio de un sistema a otro, ya que son costumbres adoptadas durante tiempo. Pienso que cuesta lo mismo o más aprender a usar Windows conociendo GNU/Linux que aprender a usar GNU/Linux conociendo Windows. Igual con otros sistemas operativos como MacOS, QNX, BeOS, AtheOS,*BSD,...
- **Como no tiene una empresa detrás, no es de fiar.** Esto es una estupidez tremenda que no voy a argumentar. Simplemente dejaré que ustedes reflexionen. ¿De quién te fiarías más: de un grupo de gente a lo largo de todo el mundo que hace algo por puro placer y lo comparte con los demás o de una empresa cuyo único interés es ganar el mayor dinero posible a tu costa?

4. Ventajas de GNU/Linux

- **GNU/Linux es seguro.** GNU/Linux sigue la filosofía de los sistemas UNIX y está basado en un robusto sistema de permisos y usuarios. Esto quiere decir que un usuario puede modificar únicamente sus propios archivos a menos que el dueño le haya dado permisos. Existe una excepción: el administrador o *root*, que es el encargado de instalar y desinstalar programas, mantener cuentas de usuarios, vigilar la seguridad del sistema... En ordenadores personales, es frecuente que el *root* sea el mismo usuario.
- **Olvidate de los virus.** Es difícil no haber oído hablar a estas alturas de lo que es un virus informático. Asumiremos que es un programa que se reproduce a sí mismo y trata de extenderse a otros ordenadores (lo que haga después es otro asunto). Pueden ser completamente inofensivos o dejar el ordenador para el arrastre.

La cuestión es que no se puede afirmar que en GNU/Linux no existan virus: es otro falso tópico, pero se acerca mucho a la realidad. En sistemas como Windows aparecen cientos (o miles) de nuevos virus cada año (ahora mismo, Windows NT y 2000 son vulnerables a unos 75000 virus) mientras que en GNU/Linux y sistemas similares se pueden contar con los dedos de la mano los virus aparecidos hasta el momento. Como ya he contado, los sistemas de permisos hacen de GNU/Linux un sistema robusto y no permiten el contagio más allá del propio usuario: si un usuario resulta infectado, la mayor pérdida serán sus archivos pero el ordenador y el sistema no sufrirán ningún daño (a menos que este sea el *root*). Por otro lado, la mayoría de administradores instalan programas obtenidos de fuentes de confianza. Esto es consecuencia directa de que el software sea libre, ya que no necesitas *copias piratas*

- **...y de los pantallazos azules.** Si has sido o eres usuario de Windows, estarás más que harto de los conocidos pantallazos azules o BSOD (*Blue Screen of Death*). En GNU/Linux esas cosas no existen. Al trabajar cada programa de manera independiente, si una pequeña parte del sistema falla no hace caer al resto.

- **...y de formatear cada X meses.** Esto tiene una sencilla explicación. Para ello hablare de las *bibliotecas* o *librerías*. Lo más logico es que si dos programas tienen que hacer algo igual (una pequeña parte de su trabajo) llamen a una librería. Por ejemplo, si tenemos varios juegos, los programadores no se dedican a reescribirlos de cero: si un juego necesita (por ejemplo) dibujar un círculo simplemente dejara que una librería ya escrita lo haga por el.

La cuestión es que en sistemas Windows, las librerías que vienen por defecto son muy escasas. Por esto, los programas tienen que instalar sus propias librerías (las conocidas .dll). El problema es que no se suelen preocupar de si esas librerías ya existían, con lo que al desinstalar el programa no quita las librerías. Esto es lo que hace que un Pentium con Windows vaya como un 486 después de instalar y desinstalar cientos de programitas.

Por el contrario, los sistemas GNU/Linux llevan un control de las librerías y programas con los sistemas de paquetes, conociendo en todo momento que es cada librería, quien la utiliza y si sigue siendo necesitada.

- **Buena comunicación entre programas.** Una de las máximas en la filosofía de diseño de GNU/Linux es la "modularidad" (por llamarlo de alguna manera). Es decir, cada programa cumple una función sencilla (o compleja, pero una) pero la cumple bien. A partir de ahí, se van "encadenando" los programas.

Por ejemplo, el comando **ls** muestra un listado completo de archivos en un directorio y el comando **grep "^d*z\$"** filtra lo que introduzcas después y muestra las líneas que comiencen con *d* y acaben en *z*. Entonces el comando **ls | grep "^d*z\$"** mostrara todos los archivos que comiencen con *d* y acaben en *z*. Este quizá no sea el ejemplo más sencillo y esto puede parecer algo extraño, pero una vez que empiezas a manejarlo se aprende fácilmente y es una herramienta *muy* potente

Glosario

Código fuente

Para explicar esto haré una similitud. Imaginemos que la informática es como la cocina. Los programas ejecutables que es lo único que los usuarios de Windows suelen tener es el plato cocinado. El código fuente sería la receta. No es lo mismo probar un plato e intentar mejorarlo que saber exactamente lo que lleva.

Núcleo

El núcleo o *kernel* de un sistema operativo es la parte central de éste. Se podría decir que es el "intermediario" que comunica a los programas que usamos habitualmente con el *hardware* o componentes del ordenador como CD-ROM, DVD, disketera,...

UNIX

Sistema operativo desarrollado por AT&T en los años 70. Éste sistema tuvo un gran éxito y muchos otros se basaron en él, con lo que un programa hecho para Linux se podría usar en un UNIX de AT&T con pocas o ninguna modificación. De ahí que hablemos de sistemas tipo UNIX.