

PREGUNTAS FRECUENTE REFERENTES A SOFTWARE LIBRE

1.¿Qué es Linux?

Cuando la mayoría de la gente utiliza el término Linux, se refieren a un sistema operativo particular, el cual consiste en un programa de administración de recursos de un computador, tal y como DOS, Windows, MacOS, entre otros.

Linux, técnicamente, es únicamente un componente de todo el sistema operativo (el núcleo), y viene acompañado por una gran cantidad de herramientas de otros proyectos sin las cuales sería completamente inútil. Entre ellas se encuentran las herramientas desarrolladas por el proyecto GNU (tan importantes que algunas personas se refieren al sistema como "GNU/Linux"), la implementación del sistema de ventanas X conocida como XFree86, el servidor web Apache, el lenguaje de programación Perl, el entorno gráfico KDE y muchísimas más.

Una particularidad de GNU/Linux es que se trata de un sistema operativo multiplataforma (funciona en PC, Mac o Atari y en otras máquinas que no son PC), multitarea (es capaz de hacer varias cosas simultáneamente), multiusuario (pueden concurrir muchos usuarios simultáneamente en la misma máquina, como si cada uno dispusiese de la máquina en exclusiva) y sobre todo se adapta a las necesidades específicas de los usuarios.

2.¿Qué es el software libre?

En primer lugar es un software donde los códigos de fabricación (el llamado "código fuente", algo así como los "planos" del programa) son conocidos y difundidos libremente. El autor del software concede a cualquiera el derecho a usar su obra, a modificarla y a adaptarla a sus necesidades específicas. Este derecho de libre uso a veces se otorga sin restricciones (licencias tipo BSD) y otras veces con la única condición de que toda mejora se distribuya con las mismas condiciones de uso que tiene originalmente y por tanto siga estando libremente disponible para todo el mundo (licencias tipo copyleft, como la GPL de la Free Software Foundation). A diferencia de lo que sucede con el software propietario cuyo único objetivo es la rentabilidad económica y no el hacer buenas herramientas, la comunidad del software libre se empeña en la búsqueda de una buena adecuación entre las necesidades y el propio producto, esto es, buscar a la vez la calidad y la eficiencia social que otorga la libertad de uso. El software libre, incluso en los casos en que se comercializa, siempre es más barato que sus equivalentes propietarios, y casi siempre acaba estando disponible gratuitamente.

3.¿Cuáles son las principales ventajas de Linux?

La principal ventaja de GNU/Linux es que los principales componentes necesarios para su funcionamiento son desarrollados y distribuidos en el modelo de software libre. Por ejemplo, el XFree86 utiliza su propia licencia de software libre mientras que Linux (el núcleo) y la librería de funciones de C son distribuidas bajo la licencia GNU GPL. Aunque no hay nada que impida el desarrollo y distribución de programas no libres que operen en GNU/Linux, su comunidad de usuarios suele prestar especial atención a la libertad del software y por eso es que han tenido especial cuidado en que no se requieran componentes no libres (propietarios) para utilizarlo.

4.¿Qué se puede hacer con GNU/Linux?

GNU/Linux se puede utilizar para proveer diferentes servicios relacionados con internet (servidor web, servidor FTP, servidor de bases de datos, proveedor de acceso, etc.) y en esto su alta estabilidad y confiabilidad pueden ser una gran ventaja.

GNU/Linux también se puede usar para manejar una estación de trabajo personal: tiene una gran variedad de ambientes visuales, algunos de los cuales lo hacen muy fácil de usar. Existe una buena cantidad de programas para GNU/Linux (la mayoría de software libre) que permiten utilizarlo para las principales tareas para las que se requiere un computador de hogar: navegar en Internet, hacer documentos de texto, trabajar en hojas de cálculo, elaborar presentaciones con gráficas, llevar un registro de actividades, manejar contabilidad e información de finanzas, escuchar música, ver vídeos, entre otros.

5.¿Que hardware se debe usar para trabajar con vídeo y audio en GNU/Linux?

Para trabajar con vídeo se recomienda tener mucha capacidad de almacenamiento en discos duros de buena velocidad y utilizar en ellos algún un sistema de archivos adecuado como XFS, ext3 o ReiserFS. Si se piensa hacer cambios de formatos (transcoding) es importante tener un procesador de alta velocidad.

Otro aspecto importante de hardware es la tarjeta de video. La mayoría de los fabricantes grandes tiene soporte para Linux. ATI, NVIDIA y MATROX son algunos que soportan activamente el uso de sus tarjetas con el kernel Linux.

6.¿Qué es una distribución?

Como las respuestas anteriores explican, hay una gran cantidad de componentes necesarios para el funcionamiento del sistema operativo. En algunos casos hay múltiples alternativas y diversas maneras de configurar el sistema, todas igualmente válidas.

Una distribución de GNU/Linux es un conjunto es una selección particular de los componentes que se utilizarán, la manera en que se configurarán y los programas complementarios que se distribuirán junto con el sistema operativo para brindar funcionalidad adicional.

Algunas distribuciones suelen tener propósitos muy particulares como brindar funcionalidad específica para artistas, llevar la mayor cantidad de funcionalidad que quepa únicamente en un diskette o dar prioridad a la seguridad del sistema, pero la mayoría suelen ser diseñadas para propósitos generales.

7.¿Cuáles son las principales distribuciones?

El siguiente es un listado de algunas de las distribuciones con una breve descripción de su funcionalidad y características distintivas:

- Debian GNU/Linux, distribución que presta especial atención a la libertad del software. Es desarrollado por un gran número de personas en muchos países y no por una compañía particular.
- Gentoo, distribución de propósito general que permite ser optimizada para el procesador particular de la máquina en que será ejecutada de manera

relativamente sencilla.

- Knoppix, distribución basada en Debian con el propósito de permitir utilizar GNU/Linux sin necesidad de instalarlo en el disco duro (cargando todos los programas desde el CD en la medida en que son utilizados).
- GNU/LinEx, distribución basada en Debian desarrollada en España por la Junta de Extremadura con el propósito de contribuir al desarrollo de la Red Tecnológica Educativa y difundir el uso del software libre en la región para eliminar barreras económicas al acceso a la tecnología.
- Mandrake Linux, distribución de propósito general, basada en RedHat.
- MkLinux, distribución creada específicamente para correr en computadores Apple (con procesadores PowerPC).
- RedHat, distribución de propósito general.
- Slackware, distribución de propósito general.
- SuSE, distribución de propósito general desarrollada en Alemania.
- Yellow Dog Linux, distribución creada específicamente para correr en computadores Apple (con procesadores PowerPC).

Existen muchas distribuciones de Linux, debido a su carácter de software libre. Si está interesado en conocer más distribuciones puede dirigirse al enlace [The LWN.net Linux Distribution List](http://TheLWN.net), el cual ofrece un listado bastante completo, incluyendo información detallada sobre cada una de ellas y una clasificación especial para cada tipo de distribución.

8.¿Cómo puedo obtener una copia de una distribución?

Hay varias maneras de conseguir un CD con una distribución de GNU/Linux.

Una forma, quizá la más sencilla, es bajar la imagen (ISO) de los CDs de la distribución de su elección y quemarla a un CD. Para hacerlo necesitará una conexión a internet con buen ancho de banda pues cada CD ocupa alrededor de 700 MB.

Un buen lugar para buscar imágenes ISO es LinuxISO.org, donde encontrará un listado de distribuciones.

Tenga en cuenta que, dado que la mayoría de distribuciones de GNU/Linux son basadas en software libre (o, cuando menos, software de libre redistribución), el proceso de descarga de una distribución de GNU/Linux a través de internet es completamente legal.

Otra opción para conseguir una distribución de GNU/Linux particular es encargarla a través de su sitio web oficial. Las ventajas de ésta opción son que probablemente recibirá documentación impresa acompañando sus CDs y, más importante, que estará contribuyendo económicamente al desarrollo de la distribución de su elección. Las desventajas son que, dependiendo de la distribución, el costo puede incrementarse un poco y que tendrá que esperar un tiempo antes de recibir sus CDs.

Una última opción es asistir a uno de los festivales de instalación de software libre organizados por los integrantes de la Comunidad de Software Libre y por otras comunidades de usuarios en el país y comprar un CD con la distribución que le interese.

9.¿Es cierto que GNU/Linux es inmune a virus?

Algunas personas afirman que GNU/Linux es inmune a virus y otros males similares muy comunes en otros sistemas por razones de diseño. GNU/Linux utiliza el esquema de permisos de Unix, lo que dificulta el daño que puede hacer al sistema la ejecución de código con propósitos negativos. En caso de correr código maligno, éste únicamente tendrá acceso al sistema con los permisos del usuario original, lo que en general le impedirá modificar los archivos correspondientes a las aplicaciones para instalarse.

Sin embargo, esto no garantiza la imposibilidad de construir virus que afecten máquinas Unix. En teoría un virus relativamente sofisticado no necesita modificar archivos del sistema: aún cuando corra con las restricciones de un usuario normal, esto debe ser suficiente para afectar su entorno (de manera que, por ejemplo, sea ejecutado cada cierto tiempo) y propagarse.

Otras personas opinan que la única razón por la que no han habido mayores casos de virus que afecten máquinas GNU/Linux es porque es mucho menos popular que las máquinas Windows. Al estar instalado en alrededor del 90% de los computadores de escritorio, Windows suele ser un sistema operativo mucho más atractivo como objetivo para las personas que escriben y diseminan virus.

Sea por razones teóricas (relacionadas con el sistema de permisos de Unix) o por razones prácticas (relacionadas con su baja entrada en computadores de escritorio), en la práctica los riesgos de contagio de virus en máquinas GNU/Linux son inexistentes.

10.¿Qué ambientes de programación basados en software libre se encuentran disponibles?

Existe una buena cantidad de ambientes de desarrollo de software libre para diferentes lenguajes.

Una alternativa, quizá la más común, consiste en utilizar un editor de texto integrado a las diversas herramientas del proyecto GNU.

Los editores de texto utilizados con mayor frecuencia para desarrollar software son vim y emacs. Ambos editores suelen ser un poco difíciles de utilizar al comienzo (sus comandos son poco intuitivos) pero muy eficientes una vez que el usuario se ha familiarizado con ellos. Ambos editores ofrecen las principales características que se pueden esperar en un editor de software moderno. Por ejemplo, tienen coloración de sintaxis para una gran cantidad de lenguajes de programación, permiten editar archivos remotamente (sobre FTP, Secure Shell u otros protocolos), entre otros.

Existe una gran cantidad de herramientas que se integran fácilmente con estos editores:

- Para compilar los archivos se suele ejecutar GCC, el compilador del proyecto GNU, capaz de compilar código en C, C++, Fortran, Java y otros lenguajes. La integración entre GCC y estos editores les permite a los últimos reportar los errores detectados en el proceso de compilación de manera eficiente.

- Para manejar programas con varios archivos de código fuente se suelen utilizar archivos Makefile, manejados por GNU Make, en los que se describen las dependencias entre los archivos. Para proyectos de gran tamaño es recomendable utilizar Autoconf y Automake, dos herramientas también del proyecto GNU, que ayudan con la generación de los archivos Makefile y a que el software funcione en la mayor cantidad de plataformas posibles.
- Para depurar código de C o C++ se suele utilizar GDB, el depurador del proyecto GNU. La integración de Emacs con GDB permite controlar el proceso de depuración enteramente desde Emacs de manera bastante práctica.
- Para ayudar a la navegación del código se suelen utilizar archivos especiales (de "tags"), que contienen una lista de todos los símbolos (funciones, variables, estructuras, etc.) definidos en todos los archivos de código que hacen parte del proyecto.

Otra alternativa diferente es utilizar un ambiente de desarrollo integrado (a éstos se les suele llamar IDEs, del inglés "integrated development environment"). Entre los IDEs disponibles en software libre se encuentran los siguientes:

- Anjuta, un ambiente de desarrollo pensado especialmente en C y C++. Fue escrito por el proyecto GNOME utilizando GTK+ y provee manejo de proyectos, guías (wizards) para guiar la creación de aplicaciones, depuración interactiva y edición de código.
- Eclipse, un avanzado ambiente de desarrollo, que suele ser muy utilizado para proyectos basados en Java. Es bastante modular y existe una gran cantidad de plugins que aumentan su funcionalidad drásticamente, probablemente convirtiéndolo en la mejor herramienta libre para trabajar J2EE.
- KDevelop, un ambiente desarrollado por el proyecto KDE. Provee manejo de proyectos (generación de archivos Makefile), integración con herramientas de traducción e internacionalización, depuración interactiva, control de versiones (apoyado en CVS), edición de código y generación de interfaces gráficas (tanto para KDE/Qt como para GNOME/GTK+). Soporta proyectos basados en C, C++, Java, Add, Perl, Ruby y Pascal.

11. ¿Pueden coexistir y compartir recursos los sistemas Linux y sistemas de Microsoft?

Sí, Linux a diferencia de los sistemas Microsoft, puede coexistir y compartir de manera homogénea los recursos informáticos existentes.

El único beneficiario de esta característica es el usuario y esto mismo es garantía de una migración a soluciones basadas en las necesidades de los usuarios.

12. ¿Por qué existen muchas distribuciones de Linux?

Una distribución de Linux es un conjunto de paquetes que se agrupan con la idea de cumplir un determinado objetivo.

Según los requerimientos, una distribución puede ser fácil, ser veloz, ser agradable visualmente, usar pocos recursos de hardware, ser amigable, ser para desarrolladores, ser para usuarios finales, aparte de que siempre existen y existirán

tantas ofertas de distribuciones como necesidades diferentes presente el usuario de los sistemas.

13.¿Qué es un proceso de migración a Software Libre?

La migración a Software Libre se refiere a un conjunto de actuaciones cuya finalidad es la sustitución de infraestructuras tecnológicas apoyadas en software propietario por otras con funciones equivalentes basadas en Software Libre.

14.¿Por qué hay que evitar el envío de documentos .doc?

Los archivos .doc son documentos generados con una aplicación propietaria. Cuando se envían archivos con esta extensión no sólo estamos obligando al receptor del programa a disponer de esa aplicación (y a pagar por ella), sino que estamos generalizando el uso de una aplicación que es propiedad de una empresa y en consecuencia trabajamos en base a los criterios de entidades privadas.

15.¿Que gana el estado con el uso total de software libre?

En muchas ocasiones se antepone delante de todas las ventajas el ahorro monetario. Según los sistemas instalados, sus costos, y las herramientas disponibles para reemplazarlas, este ahorro puede ser realmente importante, pero puede disminuir a corto plazo por los costos de realizar la transición de los sistemas. Aún así, existen muchas otras ventajas en el uso de software libre, que son inmediatas y más importantes, al punto de ser cruciales para la adopción de estas políticas por el estado:

1. Independencia tecnológica: Mediante el uso de software libre, el estado deja de tener sus sistemas controlados por una entidad externa (con frecuencia empresas extranjeras). De esta forma rompe la dependencia tecnológica que lo tiene actualmente atado y obtiene las libertades que el software libre otorga.
1. Control de la información: Esto sale directamente como consecuencia directa de las libertades del software libre. Al tener la libertad de inspeccionar el mecanismo de funcionamiento del software y la manera en que almacena los datos, y la posibilidad de modificar (o contratar a alguien que modifique) estos aspectos, queda en manos del estado la llave del acceso a la información (en vez de quedar en manos privadas).
1. Confiabilidad y estabilidad: El software libre, al ser público, esta sometido a la inspección de una multitud de personas, que pueden buscar problemas, solucionarlos, y compartir la solución con los demás. Debido a esto, y a lo que se llama "el principio de Linux" (dada la suficiente cantidad de ojos, cualquier error del software es evidente), los programas libres gozan de un excelente nivel de confiabilidad y estabilidad, requerido para las aplicaciones críticas del estado.
1. Seguridad: Este es uno de los puntos clave para el Estado. Mucha de la información que el Estado maneja puede ser peligrosa en manos incorrectas. Es por esto que es crítico que el Estado pueda fiscalizar que su software no tenga puertas de entrada traseras, voluntarias o accidentales, y que pueda cerrarlas en caso de encontrarlas; tal inspección sólo es posible con el software libre.

16.¿Que problemas puede enfrentar implementar estas políticas?

Evidentemente, una propuesta de implementación reduciría los beneficios de los vendedores de software cerrado. Es de esperar que estos, ejerzan toda la presión a su disposición para evitar que se tomen medidas que involucren migración a software

libre. Frente a ese peligro, hay que considerar que está en juego el control de la información por parte del estado, y las libertades individuales de los ciudadanos. Se debe tenerse en cuenta que una política de este tipo no discrimina en contra de productos o proveedores específicos, sino contra ciertas prácticas nocivas que involucran el control de la información del usuario por parte del proveedor. Eso es lo que motiva las restricciones de esta política, que tienen como fin establecer cualidades mínimas para garantizar los derechos de los ciudadanos, la calidad del software y la seguridad de la información.

17.¿No se perjudicaría a la industria de software?

No, porque no se implementaría a través de prohibir la industria del software, sino fijando como condición necesaria para uso de software en el estado la característica de "libre" (De la misma manera en que suelen establecerse condiciones necesarias razonables a cumplir en cualquier licitación o contratación que hace el estado). De hecho, si en "industria de software" consideramos a la industria local, esta será ampliamente beneficiada, ya que las licencias libres le otorgan al gobierno el derecho a contratar profesionales locales para modificar y adaptar sus sistemas (incluso aquellos no desarrollados localmente), y de esta forma fomentar la industria tecnológica local, la economía y el empleo. Estos beneficios que distinguen al software libre del propietario, provienen de la posibilidad de inspección y modificación libres a cualquier individuo, en vez de estar restringidos al proveedor, que puede usar esa restricción para monopolizar el soporte técnico

18.¿No es conveniente dejar que se seleccione el mejor software, sea libre o propietario?

Al Estado le conviene elegir la solución mas apta para sus necesidades. La clave del problema está en el significado de "más apta". La solución más apta no necesariamente es la solución más usada en el mercado; muchas veces la solución más exitosa lo es solamente por que la empresa que la promovió tuvo una mejor campaña publicitaria, porque atrapó al mercado en un monopolio, porque hizo una buena movida comercial, entre otras razones validas. Tampoco es necesariamente más apta la que tiene toda la funcionalidad necesaria; el Estado necesita más que eso: necesita ser independiente tecnológicamente, poder tener control sobre su propia información y poder proteger la seguridad de sus datos. Esas son algunas de las aptitudes que sólo el software libre puede otorgar, aptitudes que el estado no puede dejar de lado (si puede dejar de lado, en cambio, funcionalidad no crítica). Por ello, la solución más apta, sea cual fuere, será un programa libre; los programas no libres tienen características que los hacen completamente inutilizables para un Estado, por someterlo a riesgos importantísimos, aún cuando la solución libre que se otorga, sea ligeramente menos funcional (mientras no se trate de funcionalidad crítica), o más costosa.

19.¿No sería costosa una migración si se tiene algo que ya funciona?

Sí, lo sería. Una migración involucra costos, toma de decisiones para implementar los nuevos sistemas, mano de obra para implementar el cambio, conversión de datos, reentrenamiento del personal, y eventualmente gastos en licencias y/o desarrollo (no todo el software libre es gratis) y tiempo. Todos estos son costos fijos, que se pagan una vez. El software propietario en funcionamiento ahora, también tuvo sus costos fijos que fueron pagado y no pueden ser recuperados. Pero además de éstos, hay otros costos involucrados en el software propietario: actualizaciones permanentes,

pérdida de interoperabilidad, mantenimiento y por sobre todo, el inmenso precio que tiene para el estado la pérdida de las libertades que le garantizan el control de su propia información. Estos costos son permanentes y crecientes a lo largo del tiempo (incluso si sólo se consideran los monetarios), y tarde o temprano, superaran a los costos fijos de realizar una migración. Por lo tanto, dado que la migración, a la larga, nos beneficiará económicamente conviene llevarla a cabo lo antes posible, en vez de esperar que los costos crezcan hasta volverse incontrolables es un costo a corto plazo, pero un ahorro enorme a largo plazo. Y que además produce beneficios mayores.

20.¿No es suficiente con tener la información en formatos públicos?

Si lo que queremos es poder acceder libremente a los datos que intercambiamos con el gobierno (formularios, informes, boletines oficiales), sí. El uso de formatos abiertos corresponde al principio que dicta que el Estado debe hacer pública su información para garantizar la transparencia. Los formatos abiertos nos ayudarían en ese sentido, pero aún quedan pendiente el problema de control de la información, la seguridad nacional, y la independencia tecnológica, virtudes que se ponen en riesgo al usar software propietario, aún si éste opera sobre formatos abiertos de datos.

21.¿De donde obtendría soporte técnico y mantenimiento el gobierno?

Esencialmente, del mismo lugar que lo hace ahora. Los técnicos empleados por el gobierno, una vez recapitados, podrían seguir cumpliendo sus funciones. El software libre tiene también soporte empresarial, al igual que el propietario. Algunas empresas como IBM dan soporte a software libre y propietario, otras como Red Hat, dan soporte a software libre solamente, y otras como FreeDevelopers desarrollan software libre a pedido.

La diferencia con el software propietario, es que ahora se podría elegir libremente a quien contratar para soporte, en función de la capacitación, y cuanto quiera cobrar. De esta forma se impide la extorsión que puede realizar la empresa dueña de un software propietario, aprovechando su exclusividad sobre el soporte y mantenimiento de sus productos.

Entre las opción para elegir, en el caso del software libre, se incluirían también como posibilidades a los técnicos y empresas locales, de esta forma fomentando el desarrollo y la economía local. Otro medio disponible son los acuerdos con las universidades, que son fuente de personal capacitado y que mediante acuerdos pueden colaborar para ofrecer soluciones y desarrollo de sistemas.

22.¿El costo no sería demasiado alto dada la enorme cantidad de código desarrollado por el estado?

Es cierto que mucho código (sobre todo administrativo) está desarrollado internamente, y los problemas de reemplazarlo son complejos, ya que no tienen reemplazo libre por ser algo muy específico, y son muy costosos para re-desarrollar. Pero dado que el código desarrollado internamente es propiedad del estado, este puede tramitar su relicenciamiento para declararlo software libre. De esta forma, todo este código se transformaría a programas libres sin esfuerzo técnico, sin necesidad de actualizar sistemas, ni volver a capacitar al personal.

23.¿No es necesario ocultar el código en ciertas áreas? ¿La publicidad

del código no facilita el acceso indebido a los criminales informáticos?

El software de seguridad (que es el que está en discusión en esta pregunta) es como un seguro de caja fuerte: aunque se sepa como funciona, es necesario conocer la "clave" o "combinación" que su dueño fijó para abrirla. La seguridad depende de la protección de esa combinación, no del mecanismo en sí, siempre y cuando el mecanismo sea lo suficientemente bueno.

Hay programas libres para usar los mecanismos de seguridad más fuertes conocidos. El hecho de que sean libres les da una garantía de calidad, ya que su publicidad permite que cualquiera pueda detectar y reparar los fallos y riesgos a la seguridad que contenga. Cuando se oculta el funcionamiento, sólo aquellos que tienen intenciones de vulnerar esta seguridad se toman el trabajo de desarmarlo y ver como funciona, aumentando el riesgo. En resumen: es posible tener programas libres de máxima seguridad, y es más fácil controlar que funcionen correctamente y auditarlos.

24.¿Forzar una política de uso de software libre no es equivalente a forzar el uso de un producto determinado?

No es comparable el software libre a una marca determinada, ya que consiste para que un programa sea libre basta con que se otorguen las facultades apropiadas al usuario, condición que cualquier empresa nacional o extranjera puede cumplir. Esto es diferente a decir "se exige marca X", que es una condición que solo la empresa X puede cumplir. Además no hay imposición sobre la libertad de decisión de los ciudadanos, ya que una política de uso exclusivo de software libre es una decisión del Estado y para el Estado, es decir, de administración de sus sistemas internos. Es lo que le corresponde a un gobierno: organizarse internamente de la mejor forma posible para defender los derechos de los ciudadanos y proteger su propia seguridad.

25.¿Es suficiente aplicar esta política sólo a los nuevos sistemas?

No, la política debe ser total y extenderse a todos los sistemas, nuevos y viejos; ya que los sistemas anteriores seguirían siendo perjudiciales en las formas mencionadas anteriormente, es decir, poniendo en riesgo la Seguridad Nacional, generando dependencia tecnológica, y impidiéndole al estado ejercer control sobre su propia información.