

9A. PARTE!

DOCUMENTAÇÕES

✓ *Copyright (c) 2002-2005 – Ednei Pacheco de Melo.*

Permission is granted to copy, distribute and/or modify this document under the terms of the *GNU Free Documentation License*, version 1.1 or any later version published by the *Free Software Foundation*; a copy of the license is included in the section entitled “*GNU Free Documentation License*”.

| | |
|---|-----------|
| ABERTURA..... | 4 |
| I. NORMAS TÉCNICAS E DE PADRONIZAÇÃO..... | 5 |
| <i>Introdução.....</i> | 5 |
| <i>Linux Standard Base.....</i> | 5 |
| Objetivos (e vantagens) da LSB..... | 5 |
| Requerimentos da LSB..... | 6 |
| <i>Sistema de Arquivos.....</i> | 6 |
| <i>Compatibilidade binária.....</i> | 6 |
| <i>Gerenciamento de programas.....</i> | 7 |
| <i>Linha de comando.....</i> | 7 |
| <i>Níveis e métodos de inicialização.....</i> | 7 |
| <i>Variáveis de ambiente.....</i> | 8 |
| <i>Bibliotecas e aplicações disponíveis.....</i> | 8 |
| <i>Kernel e respectivos módulos.....</i> | 8 |
| <i>Documentação técnica.....</i> | 9 |
| <i>Outros.....</i> | 9 |
| Beneficiados..... | 9 |
| A Free Standard Group..... | 9 |
| <i>Sobre a FHS.....</i> | 9 |
| <i>Conclusão.....</i> | 10 |
| II. LEIS, NORMAS E LEGISLAÇÕES VIGENTES..... | 12 |
| <i>Introdução.....</i> | 12 |
| <i>O Software Livre.....</i> | 12 |
| O Projeto GNU e a Free Software Foundation..... | 12 |
| <i>Diferenças entre os softwares livres e proprietários.....</i> | 13 |
| Uso..... | 13 |
| Integridade..... | 13 |
| Reprodução..... | 14 |
| Garantia e suporte..... | 14 |
| <i>Modalidades de licenciamento.....</i> | 14 |
| Copyleft x Copyright..... | 14 |
| <i>As licenças do Projeto GNU.....</i> | 15 |
| GPL..... | 15 |
| LGPL..... | 15 |
| FDL..... | 16 |
| <i>Sobre as traduções extra-oficiais das licenças.....</i> | 16 |
| <i>Razões para o uso e disseminação do Software Livre.....</i> | 17 |
| A pirataria de software..... | 17 |
| Como a pirataria prejudica a disseminação do Software Livre..... | 17 |
| O conceito fair use (uso justo)..... | 18 |
| <i>Sobre outras licenças livres.....</i> | 19 |
| <i>Conclusão.....</i> | 19 |

| | |
|--------------------------------------|-----------|
| III. A LPI..... | 20 |
| <i>Introdução.....</i> | <i>20</i> |
| <i>O que é?.....</i> | <i>20</i> |
| Patrocinadores..... | 20 |
| Reconhecimento..... | 21 |
| <i>A certificação.....</i> | <i>21</i> |
| Modalidades..... | 21 |
| Vantagens..... | 21 |
| Cadastro..... | 21 |
| Calendário..... | 22 |
| <i>Conclusão.....</i> | <i>23</i> |
| IV. TEXTOS INTERESSANTES..... | 24 |
| <i>Introdução.....</i> | <i>24</i> |
| <i>Os textos.....</i> | <i>24</i> |
| Projeto GNU..... | 24 |
| Free Software Foundation..... | 25 |
| Outros..... | 25 |
| <i>Conclusão.....</i> | <i>25</i> |
| ENCERRAMENTO..... | 26 |

Enfim, chegamos então à última parte deste livro que, conforme intitulado, trata-se das documentações gerais sobre as obras existentes no mundo do *Software* (e documentação) *Livre*, além das especificações técnicas para a padronização dos aplicativos e distribuições. Não ignorem esta parte, ao considerarem que as informações contidas nos próximos capítulos não mereçam a mesma atenção dada às demais informações discutidas durante todo o livro. Ao contrário do que a grande maioria dos usuários pensam (e até mesmo nem fazem idéia), o conhecimento das questões referentes à normas de padronização e licenciamento das obras que enriquecem o mundo dos *softwares* livres são de extrema importância para a compreensão de suas particularidades.

Por quê agora recentemente – mais precisamente de 1999 em diante –, após mais de 15 anos, a filosofia de liberdade, mútua colaboração e partilha de conhecimentos, têm tido um importante papel para os rumos da tecnologia de informação? De onde “*saí*” esta sua incrível flexibilidade? Por que é dada uma grande importância ao crescimento e adoção dos sistemas *GNU/Linux* nas mais diversas áreas tecnológica? E o por quê da existência de distintas posições das grandes corporações – umas apóiam fervorosamente o movimento de código-aberto; outras atacam a imagem desta maravilhosa ideologia que originou este incrível sistema operacional? Por quê será?

Estas e outras questões possuem respostas tão variadas quanto o ponto de vista dos especialistas, que por sua vez seria praticamente impossível responder cada uma delas dado aos seus detalhes e assunto acerca relacionados – e até mesmo perigoso, dada às complexas questões da legislação vigente –, no que resultaria em escrever praticamente outro livro, ao invés de alguns simples capítulos.

Ao invés de respondermos estas questões e muitas outras – que por sua vez resultaria em inúmeras e diferentes opiniões –, optamos por desenvolver uma parte em especial – *Documentações* – para que os leitores conheçam os princípios básicos das leis, normas e especificações vigentes dos sistemas *GNU/Linux* e do movimento do *Software Livre*, além de terem referências para acessar na íntegra estas documentações. Nos próximos capítulos também encontraremos alguns comentários interessantes como acerca das traduções como a licença *GPL* (extra-oficial), além de textos interessantes do *Projeto GNU* para melhor compreensão. &;-D

I. NORMAS TÉCNICAS E DE PADRONIZAÇÃO

INTRODUÇÃO

Uma das maiores dificuldades enfrentadas pela grande maioria dos usuários e empresas que utilizam o sistema *GNU/Linux* é a questão da “falta de padronização” entre as diferentes distribuições existentes. Isto se deve ao fato da existência de seus conceitos e idéias próprias e diferenciadas. Muitos utilitários especiais se encontram presentes somente em algumas, noutras existem diversas estruturas diferenciadas, além de muitas outras mudanças para a sua própria satisfação. Face aos problemas causados por estas diferenças, foram desenvolvidos um conjunto de normas e especificações técnicas que visam definir padrões para minimizar a incompatibilidade existente entre as distribuições.

Existem diversas normas e especificações técnicas, todas com os mesmos objetivos de compatibilidade e padronização, sendo que destas, destaca-se a *Linux Standard Base*.

LINUX STANDARD BASE

✓ <<http://www.linuxbase.org/>>.

Antigamente, devido ao auxílio de diversos voluntários espalhados no mundo inteiro para o seu desenvolvimento, as distribuições *GNU/Linux* utilizavam dos mais variados recursos de desenvolvimento e implementação com uma liberdade tal que em diversos aspectos tornavam-se distintas e até mesmo incompatíveis. Isto gerou enormes dificuldades de utilização à tal ponto que houve uma grande convergência de padrões, onde posteriormente a comunidade, junto com grandes corporações defensoras dos sistemas *GNU/Linux*, resolveram criar um conjunto de novos padrões para (pelo menos) minimizar este problema. À partir daí surgiu então a *LSB* – *Linux Standard Base* – um conjunto de normas e padrões técnicos que visa obter ou melhorar a compatibilidade entre as distribuições *GNU/Linux* existentes no mercado.

OBJETIVOS (E VANTAGENS) DA LSB

Além da difícil meta de padronização conforme dito anteriormente, dentre os outros principais objetivos almejados pela *LSB*, destacam-se:

- Tornar todos os aplicativos aptos à serem suportados pelas distribuições que encontram-se em conformidade com os padrões pré-estabelecidos por ela; na prática poderemos utilizar as aplicações da distribuição *X* na distribuição *Y*, desde que obedeçam os critérios acima especificados;
- Facilitar a vida dos profissionais que lidam com os sistemas

GNU/Linux, evitando maiores dificuldades de se adaptarem em virtude das diferentes implementações de recursos técnicos;

- Prover maior segurança, agilidade e redução de custos às empresas que optam por manter estações de trabalho com sistemas *GNU/Linux*, em virtude do alto custo e baixa qualidade do suporte técnico – outra das grandes barreiras que impedem a sua adoção;
- Auxiliar ao desenvolvimento de materiais técnicos e didáticos, que atualmente focam uma única ou um pequeno conjunto básico de distribuições específicas.

Ao contrário de muitos boatos e rumores, a *LSB* somente estabelece um núcleo de recursos e capacidades que deverão ser adotados pelas distribuições que desejam a certificação, e **NÃO** como estas deverão ser construídas. Na prática, todos estes sistemas continuarão tendo a liberdade de implementar suas melhorias conforme melhor considerarem, porém, deverão seguir algumas normas e padrões técnicos pré-determinados.

REQUERIMENTOS DA LSB

Dentre alguns requerimentos da *LSB*, destacam-se as seguintes categorias para que uma distribuição seja certificada pelo padrão:

SISTEMA DE ARQUIVOS

As especificações da *LSB* referentes ao sistema de arquivos são regidas pela *FHS*. Uma distribuição, para estar dentro dos padrões estabelecidos pela *LSB*, deverá obedecer essencialmente os critérios adotados pela *FHS*. Consulte neste capítulo a seção *Filesystem Hierarchy Standard* para obter maiores informações.

COMPATIBILIDADE BINÁRIA

Infelizmente dada as diferenças entre as distribuições, muito provavelmente um determinado pacote ou conjunto de pacotes compilados especificamente para uma distribuição não poderão funcionar em outras ou funcionar precariamente, pelo menos na grande maioria das vezes. Isto ocorre em virtude da implementação de diferentes versões dos compiladores e bibliotecas padrões do sistema, como o *GCC* e a *libc*. Inicialmente parece ser um problema de pouca importância quando não só encontramos à disponibilidade a maioria dos programas necessários como também os códigos-fontes disponíveis na página oficial do projeto.

Porém em diversas circunstâncias serão necessários a utilização de aplicações proprietárias, como as soluções disponibilizadas pela *Borland* ou *Sun*, por exemplo. Este é o principal motivo pelo qual um determinado programa se encontra disponível apenas para distribuições específicas e ainda para uma determinada versão ou a partir desta.



A *LSB* provê uma série de requerimentos que visam diminuir a incompatibilidade binária entre as distribuições, tendo como objetivo a possibilidade de desenvolvimento de pacotes binários que possa ser utilizado por diferentes distribuições. Tanto para pacotes específicos de uma determinada distribuição quanto para programas comerciais. No caso da biblioteca *glibc*, felizmente os sistemas *GNU/Linux* permitem a manutenção de diferentes versões desta biblioteca no sistema, o que ameniza bastante esta situação.

GERENCIAMENTO DE PROGRAMAS

O gerenciador padrão certificadas pela *LSB* será o *RPM*, por se tratar de um formato pioneiro e universal entre muitas distribuições.

Esta medida é importante especialmente pelo fato de diversas aplicações – especialmente as proprietárias – não utilizarem as ferramentas de gerenciamento de programas nativo do sistema. Estes desenvolvedores geralmente optam por utilizar outros métodos – como um assistente de instalação – que muitas vezes impossibilita um melhor gerenciamento dos programas instalados no sistema.

Um detalhe importante para observar é que a *LSB* não define o gerenciador de pacotes padrão do sistema, e sim qual deverá estar implementado para uma maior compatibilidade geral. Para as outras distribuições que não utilizam o *RPM* (como o *Slackware*, *Debian* e *Stampage*), estas poderão manter seus formatos de arquivos nativos, desde que possuam este utilitário no sistema para possibilitar a instalação de pacotes *.rpm*.

LINHA DE COMANDO

Por ser essencial para a boa administração do sistema, um dos principais requerimentos da *LSB* será a adoção do *Bash* como o interpretador de comandos oficial das distribuições certificadas devido à sua grande difusão entre as distribuições, além dos recursos que o mesmo oferece.

NÍVEIS E MÉTODOS DE INICIALIZAÇÃO

Os métodos de inicialização deverão corresponder as especificações adotados pelo *System V*. Pelo fato do *Slackware* utilizar o *layout* do *BSD*, é exigida da distribuição o condicionamento de seus *scripts* para que se tornem compatíveis ao padrão. Felizmente isto se encontra em vigor na distribuição, pois a mesma disponibiliza o *script* */etc/rc.d/rc.sysvinit* que provê compatibilidade com serviços preparados para funcionar com inicialização *System V*.

```
#!/bin/sh
#
# rc.sysvinit This file provides basic compatibility with SystemV style
#             startup scripts. The SystemV style init system places
#             start/stop scripts for each runlevel into directories such as
#             /etc/rc.d/rc3.d/ (for runlevel 3) instead of starting them
```



```
#      from /etc/rc.d/rc.M. This makes for a lot more init scripts,
#      and a more complicated execution path to follow through if
#      something goes wrong. For this reason, Slackware has always
#      used the traditional BSD style init script layout.
#
#      However, many binary packages exist that install SystemV
#      init scripts. With rc.sysvinit in place, most well-written
#      startup scripts will work. This is primarily intended to
#      support commercial software, though, and probably shouldn't
#      be considered bug free.
#
#      Written by Patrick Volkerding <volkerdi@slackware.com>, 1999
#      from an example by Miquel van Smoorenburg
<miquels@cistron.nl>.
-//-
```

Para obterem maiores informações, consultem a *2a. Parte: Conhecimentos Gerais* -> *A inicialização*.

VARIÁVEIS DE AMBIENTE

Dentre outras especificações, deverão constar variáveis para:

- Compiladores do sistema;
- Atalhos;
- Carregamento dinâmico e especificações;
- Bibliotecas;
- Entre outros.

Todas as variáveis deverão ser especificadas principalmente em */etc/profile*.

BIBLIOTECAS E APLICAÇÕES DISPONÍVEIS

Todas as distribuições deverão ter um conjunto de bibliotecas, aplicações e arquivos de configuração disponíveis para as atividades mais realizadas. Com isto serão evitados diversos inconvenientes, como a falta de determinadas pendências para a instalação de outras aplicações.

KERNEL E RESPECTIVOS MÓDULOS

O *kernel* e respectivos módulos deverão se encontrar previamente compilados para suportar determinadas tecnologia e dispositivos de sistema, com o intuito de facilitar e padronizar a implementação de serviços que necessitem destas em diferentes distribuições sem que haja a necessidade de realizar intervenções específicas pela falta deste suporte ou pelo fato de uma distro não a possuir.



DOCUMENTAÇÃO TÉCNICA

A documentação técnica de cada pacote instalado no sistema deverão se encontrar arquivados no diretório padrão (*/usr/doc*) no seguinte formato:

```
/usr/doc/[FORMATO]/[PACOTE]/[VERSÃO]
```

Praticamente todas as distribuições armazenam as documentações dos programas neste diretório, porém algumas diferem na sua organização.

OUTROS

- Gerenciamento de usuários / autenticação / segurança;
- Definições de línguas e aspectos regionais;
- Especificações técnicas de arquitetura;
- Sistema de contagem de datas;
- etc.

Lembrando que o padrão e suas especificações estão em constantes aprimoramentos, onde uma das recomendações que deixamos é consultar a página oficial e estar atento às melhorias realizadas.

BENEFICIADOS

Os maiores beneficiados com a padronização da *LSB* são todos os profissionais que trabalham diretamente ou indiretamente com sistemas *GNU/Linux*, que vão desde simples usuários à administradores de sistemas, administradores de redes, programadores e até mesmo os profissionais que escrevem toda as documentações técnicas e didáticas.

Para nós, usuários comuns, a principal vantagem é a possibilidade de utilizarmos programas de diversos formatos e distribuições, além de ter maiores facilidades da manutenção do sistema, pelo fato dos arquivos de configuração e respectivas documentações estarem padronizadas, atendendo assim a diversas (e diferentes) distribuições.

A FREE STANDARD GROUP

A *LSB* atualmente é desenvolvida com a participação de projetos como o *Debian* e de empresas renomadas, como a *Red Hat*, *Suse*, *Conectiva* e *Caldera*, além de outros gigantes da tecnologia como a *HP*, a *Compaq (Digital)*, a *IBM*, a *SGI*, a *Sun*, entre diversos outros, onde formam a *Free Standard Group*, além de contar com apoio do próprio *Linus Torvalds*.

SOBRE A FHS

✓ <http://www.pathname.com/fhs/>.



A norma *FHS – Filesystem Hierarchy Standard* – é um conjunto de requerimentos técnicos que visam estabelecer normas e padrões para a estrutura do sistema de arquivos *Unix*, derivados e clones.

Segue abaixo um simples resumo dos principais diretórios que compõem a estrutura do sistema e seus respectivos conteúdos, para que se possa ter uma noção básica.

| <i>Filesystem Hierarchy Standard</i> | |
|---|--|
| <i>/bin</i> | Arquivos binários com os comandos essenciais para os usuários. |
| <i>/boot</i> | Arquivos estáticos para a inicialização de sistema. |
| <i>/dev</i> | Arquivos <i>devices</i> para o acesso a dispositivos. |
| <i>/etc</i> | Configurações para a máquina local. |
| <i>/home</i> | Arquivos pessoais dos usuários. |
| <i>/lib</i> | Bibliotecas compartilhadas e módulos. |
| <i>/mnt</i> <i>/media</i> | Ponto de montagem para acesso a diversos dispositivos de dados. |
| <i>/opt</i> | Manutenção de compatibilidade com antigos programas. |
| <i>/proc</i> <i>/sys</i> | Sistema de arquivo virtual com informações de processos do núcleo do sistema – <i>kernel</i> . |
| <i>/root</i> | Diretório do superusuário. |
| <i>/sbin</i> | Arquivos binários essenciais para a administração. |
| <i>/tmp</i> | Armazenamento de arquivos temporários. |
| <i>/usr</i> | Aplicativos e utilitários do sistema. |
| <i>/var</i> | Informações variáveis. |

A última versão do *FHS* é a 2.2, datada em 24 de maio de 2001, onde sua aprovação foi demorada em consequência da não concordância e de vários desentendimentos entre as principais distribuições.

Para obter maiores informações sobre os principais diretórios da raiz do sistema, diversos aspectos interessantes e informações adicionais, consulte a 3a. Parte: *Conhecimentos Gerais -> Sistema de Dados*.

CONCLUSÃO

Tão importante quanto desenvolver soluções, é torná-las compatíveis e padronizadas para o seu uso e manutenção, pois durante muito tempo estas eram as grandes dificuldades para se desenvolver e utilizar diversos programas em sistemas *GNU/Linux*, face a falta de normas e padrões rígidos e determinados. Quem nunca teve algum descontentamento ao instalar um pacote no formato *RPM* de uma distribuição em outra e seu

respectivo aplicativo não funcionar no novo ambiente?

Hoje, o universo mudou; existem atualmente um conjunto de normas e procedimentos técnicos que visam tornar os sistemas *GNU/Linux* compatíveis e interoperáveis entre si, tornando ainda mais atraente e fascinante o mundo do *Software Livre* para o mercado. &;-D



II. LEIS, NORMAS E LEGISLAÇÕES VIGENTES

INTRODUÇÃO

Padrões, normas, leis, termos... o que isto têm de importante no universo dos programas de código-aberto? A idéia não é ter a tão desejada liberdade? Exato! E tão importante quanto ter a liberdade, é garanti-la. As licenças aplicadas para os programas e as documentações tradicionais consistem principalmente em restringir a distribuição e redistribuição da obra; porém as licenças de programas e documentações livres têm um caráter diferenciado, que é basicamente a preservação do direito e da liberdade dos indivíduos que a utilizam e colaboram.

Por diversos meios de comunicação ouvimos muitas vezes a falar sobre questões de pirataria, propriedade intelectual, patentes e muitos outros assuntos pertinentes, porém estes mesmos meios não nos esclarece as limitações e artimanhas impostas pelo uso de produtos proprietários e patenteados. Neste capítulo teremos uma noção acerta destes assuntos, tendo o ressaltado os conceitos e vantagens da adoção do *Software Livre*.

O SOFTWARE LIVRE

Um programa proprietário caracteriza-se por ser licenciado com leis e normas de *Copyright* que protege a obra e/ou o seu conteúdo, restringindo e/ou proibindo toda e qualquer prática de disseminação não prevista pela licença ou contrato. O usuário tendo o licenciamento do programa apenas possui o direito de utilizá-lo para fins particulares, sendo restringido de diversas práticas especificadas nos parágrafos que compõe o documento.

Já o *Software Livre* livre possui conceitos distintos do sistema proprietário, onde o seu diferencial se encontra justamente em seu conceito de liberdade: um programa livre são programas de computadores como quaisquer outros. Podem ser editores de textos, utilitários, ferramentas de programação, etc. O que os diferem do proprietário é a possibilidade copiar, modificar e redistribuir a obra, devendo preservar os mesmos direitos – ou não – aos terceiros, de acordo com os termos da licença.

Infelizmente a grande maioria vêem o custo como a principal – e muitas vezes a única – marcantes característica e qualidade. Existem diversas outras importantes para que um programa possa ser considerado livre.

O PROJETO GNU E A FREE SOFTWARE FOUNDATION

✓ <<http://www.gnu.org/home.pt.html>>.

Conforme já explicado no capítulo “*Os sistemas GNU/Linux*”, no início dos anos 80, Richard Stallman fundou o *Projeto GNU* e a *FSF – Free Software Foundation* – com o objetivo de desenvolver um sistema operacional e um

conjunto de ferramentas livre com a ajuda de milhares de programadores, além de disseminar a ideologia do *Software Livre*.



O GNU, símbolo do projeto GNU.

"A Fundação para o Software Livre (FSF) é dedicada à eliminação de restrições sobre a cópia, redistribuição, entendimento e modificação de programas de computadores. Nós fazemos isso promovendo o desenvolvimento e o uso de software livre em todas as áreas da computação -- mas particularmente, ajudando a desenvolver o sistema operacional GNU." -> [Richard Stallman, <<http://www.gnu.org/fsf/fsf.pt.html>>].

Visite a página oficial do projeto, onde encontraremos diversos textos e artigos que ilustram bem todos estes ideais em:

DIFERENÇAS ENTRE OS SOFTWARES LIVRES E PROPRIETÁRIOS

Uso

No *software* proprietário, somente o usuário detentor da licença de utilização é que poderá realizar o uso do *software* adquirido, ou de acordo com os termos pré-estabelecidos em sua licença. Para algumas circunstâncias, obtém-se uma licença para múltiplos usuários. Já no *Software Livre* não existe nenhuma restrição de licenciamento, seja para qual finalidade for.

INTEGRIDADE

Os programas proprietários não podem ser modificados; pelo fato do código-fonte não se encontrar disponível, é impossível condicioná-los para atenderem à propósitos específicos, como também são coibidas quaisquer técnicas de decodificação para conversão do código binário em fonte, os quais os infratores poderão ser punidos por estas práticas.

O *Software Livre* possui características bem interessantes sobre este aspecto. De acordo com os termos da licença, é obrigatório a concessão do código-fonte junto a programa para que todos possam usufruir do direito de



liberdade. Com isto, poderemos editar, adaptar, customizar, e redistribuir tanto o código-fonte original quanto o alterado, mantendo os mesmos termos de licenciamento conforme a *GPL*.

REPRODUÇÃO

Um dos principais termos redigidos em uma licença de programas proprietários quanto à questão da reprodução está na expressa proibição deste ato, além de contemplar também para a sua redistribuição. O infrator fica sujeito à penalidades como multa no valor de 500 a 2.000 vezes o valor da licença e reclusão de até 6 meses. Já na modalidade *Software Livre* – como o próprio nome diz – poderemos reproduzir e redistribuir quaisquer programa regidos pelos seus termos.

GARANTIA E SUPORTE

Em virtude da liberdade que os desenvolvedores possuem para a modificação do código-fonte, bem como a impossibilidade de ter controle sobre a distribuição, os autores não são obrigados dar garantias e à assumir responsabilidades destes produtos. O mesmo se dá quando da necessidade de suporte, à salvo quando são cobrados serviços de prestação de suporte.

MODALIDADES DE LICENCIAMENTO

Diferente dos programas proprietários, não existem variações de seu licenciamento e forma de distribuição, como são feitos nas licenças de *Shareware* (demonstração), *Trial* (com validade), *Freeware* (gratuito), *Abandoware* (abandonado – que não quer dizer necessariamente livre para uso). Apenas temos a licença *GNU LGPL*, que se destina à algumas necessidades específicas, porém sem perder o espírito de liberdade.

Já na licença *GNU FDL* temos a liberdade de especificar em uma seção à parte, determinados requerimentos para a documentação à ser distribuída. Poderemos definir se esta deverá ser disponibilizada por completo ou em partes, ou ainda se determinadas partes poderão ser editadas ou não – chamadas *seções invariantes*.

COPYLEFT X COPYRIGHT

Conforme as próprias definições do *Projeto GNU*, *copyleft* é um método legal de tornar um programa em *Software Livre* e exigir que todas as versões modificadas e extendidas do mesmo também recebam o mesmo licenciamento. Traduzindo em miúdos, significa que poderemos realizar a cópia, a alteração e distribuição destes programas, porém o trabalho derivado somente poderá ser distribuído como *Software Livre*, obedecendo as regras da licença *GNU GPL*. Mas o que o *copyright* tem a ver com isso?



Para tornar um programa livre pelos termos da *GNU GPL* – ou seja, *copyleft* – inicialmente será necessário o seu registro autoria como *software copyright*, para que o desenvolvedor adicione os termos legais específicos, tornando o *software* se torne *copyleft* e assim garantir a sua liberdade de uso.

A finalidade do *copyright* é eliminar o livre uso e redistribuição das aplicações por ela protegidos, além da autoria do *software*; já o *copyleft* tem o efeito invertido: o objetivo é de garantir a liberdade de uso e redistribuição das aplicações dentro dos termos da *GNU GPL*.

AS LICENÇAS DO PROJETO GNU

✓ <<http://www.gnu.org/licenses/licenses.pt.html>>.

O *Projeto GNU* e a fundação *FSF* são os pilares do movimento do código-aberto. Se não fosse a iniciativa do líder *Richard Stallman*, de seus inúmeros colaboradores e muitas outras iniciativas, estaríamos hoje expostos à limitada escolha de programas e sistemas operacionais, sofrendo ainda as abusivas cláusulas das licenças de caráter proprietária.

Além da construção de diversas ferramentas para a concepção de um novo sistema operacional, *Richard Stallman* desenvolveu a *GPL* e posteriormente a *LGPL* e *FDL*, que são licenças especiais que visam proteger a liberdade de edição e redistribuição dos programas de código livre e documentações que fossem licenciados nestas modalidades.

GPL

✓ <<http://www.gnu.org/licenses/gpl.txt>>.

Conforme a própria definição de sua página oficial, a *Licença Pública Geral do GNU* é freqüentemente chamada abreviadamente de *GNU GPL* e é utilizada pela maioria dos programas do *Projeto GNU* assim como muitos outros programas de código aberto que não são parte do *Projeto GNU*.

A *Licença Pública Geral* não permite que incorporem o código de programas por ela licenciados em programas proprietários. Se o programa em questão é uma biblioteca de sub-rotinas, deveremos considerar mais útil permitir ligar aplicações proprietárias com a biblioteca. Se isto é o que nós desejamos, deveremos utilizar então a *Licença Pública Geral de Bibliotecas GNU*, ao invés da tradicional *GNU GPL*.

LGPL

✓ <<http://www.gnu.org/licenses/lgpl.txt>>.

Antigamente esta licença era chamada de “*Licença Pública Geral para Bibliotecas*”, popularmente chamada de “*GPL de Bibliotecas*” - mais especificamente entre as versões 2. e 2.1 -, porém devido à esta



nomenclatura, esta licença era utilizada com mais frequência do que realmente deveria ser. Seu novo nome foi criado então para que os usuários utilizem-na realmente no seu verdadeiro propósito.

Diferente da *GNU GPL*, a *Licença Pública Menos Geral do GNU* consiste em uma licença específica para o uso apenas em circunstâncias onde há necessidade de se ligar a um programa ou biblioteca proprietária sem que os termos de liberdade alterem a licença desta última. Portanto, tal programa ou biblioteca continuam sendo proprietários.

FDL

✓ <<http://www.gnu.org/licenses/fdl.txt>>.

Conforme as definições do *Projeto GNU*, a *Licença de Documentação Livre do GNU* é uma forma de *copyleft* criada para uso em manuais, livros texto ou outros documentos para garantir que qualquer um tem a real liberdade de copiar e redistribuí-los, com ou sem modificações, tanto comercial quanto não-comercialmente, desde que atenda à alguns requisitos sobre a questão de autoria por elas solicitados.

Da mesma forma que a *GPL* é largamente utilizada para os *softwares* livres, a *FDL* está no mesmo nível para as documentações livres.

SOBRE AS TRADUÇÕES EXTRA-OFICIAIS DAS LICENÇAS

✓ <<http://www.gnu.org/copyleft/copyleft.pt.html>>.

Conforme as instruções contidas na página oficial do *Projeto GNU*, somente as versões em inglês das licenças do *Projeto GNU* especificam os termos reais da distribuição dos *softwares* de código-aberto por elas licenciadas, e por isto é exigido unicamente a sua utilização para efeitos legais. Isto ocorre em virtude da dificuldade e falta de recursos para realizar a tradução exatamente dentro dos termos da licença original. A ocorrência de quaisquer erro de interpretação ocasionada por emprego de termos que não exprimem exatamente o significado da licença, poderá ocasionar sérios problemas de legislação para a comunidade de *Software Livre*, e conseqüentemente recair em diversos prejuízos para a disseminação da ideologia. Mas para auxiliar os usuários a ter um melhor entendimento de seus conceitos, somente é permitida nestes circunstâncias a publicação destas traduções em caráter extra-oficial, onde para isto deverão constar no corpo da tradução a seguintes notas:

Oficial (em inglês):

*"This is an unofficial translation of the GNU General Public License into **language**. It was not published by the Free Software Foundation, and does not legally state the distribution terms for software that uses the GNU GPL--only the original English text of the GNU GPL does that. However, we hope that this translation will*



help language speakers understand the GNU GPL better."

Traduzida (português):

*"Esta é uma tradução extra-oficial da Licença Pública Genérica do GNU para o idioma **linguagem**. Ela não foi publicada pela Fundação para o Software Livre, e não determina juridicamente os termos de distribuição para o software que utiliza a GNU GPL -- somente o texto original em inglês da GNU GPL faz isso. Entretanto, nós esperamos que esta tradução ajude pessoas que falam idioma a entender a GNU GPL melhor."*

Onde *laguange* e *linguagem* são deverão ser substituídos pela tradução dos termos na língua nativa: *portuguese* (em inglês) e *português*.

Infelizmente para o *Brasil*, apenas temos à disposição a tradução extra-oficial das licenças *GNU GPL* e *FDL*, as quais são válidas apenas para melhor entendimento de seus termos.

RAZÕES PARA O USO E DISSEMINAÇÃO DO SOFTWARE LIVRE

Existem diversas razões para o uso e disseminação do *Software Livre*, porém a principal delas é a LIBERDADE de uso do *software*. Para entender melhor, consultem neste capítulo, a seção *Textos interessantes do Projeto GNU*. Lá encontraremos endereços que apontam para textos com informações informações que traduzem com exatidão a razão para o uso e disseminação do *Software Livre*.

A PIRATARIA DE SOFTWARE

Vejamos um simples exemplo prático: o empréstimo de um *CD-ROM* de instalação de um programa qualquer à um amigo ou colega que solicita. Ambos estão violando os termos de licenciamento da obra que refere-se à reprodução destinado apenas para o detentor destes direitos, já que o solicitante não a possui. Outra violação está no fato da redistribuição, pois nada poderá impedir que este amigo faça uma cópia da obra emprestada, que por sua vez tem o consentimento do dono, já que o emprestou.

Porém, tanto pelas práticas comerciais abusivas de muitas desenvolvedoras de *software* como pelo descompromisso dos utilizadores da obra, as restrições impostas pelas licenças de propriedade intelectual são pouquíssimas vezes aplicáveis, o que resulta nas infrações conhecidas popularmente como pirataria.

Oficialmente a pirataria de *software* caracteriza-se pela utilização de um programa sem prévia autorização do autor, ou a utilização do programa licenciado em si, porém fora das normas e restrições que à compõe.

COMO A PIRATARIA PREJUDICA A DISSEMINAÇÃO DO SOFTWARE LIVRE

Ao contrário do que muitos pensam, a pirataria de *Software* é muito

prejudicial para o *Software Livre* por diversos motivos. O principal deles está em sua disseminação, já que o fator custo é o principal motivo pelo qual os usuários e as empresas adotam os sistemas e aplicações livres. E também é justamente o fator custo que levam os sistemas proprietários à serem prejudicados com a prática da pirataria, dado o alto custo de suas licenças.

Os efeitos prejudiciais da pirataria afetam:

1. A disseminação da ideologia;
2. A redistribuição dos *softwares*;
3. A implementação de melhorias;
4. Entre outros.

Quantos usuários deixaram de obter *softwares* originais na compra de computadores novos para a redução de custo, onde posteriormente realizaram a instalação destes, porém sem licença? Quantas empresas deixaram de utilizarem opções alternativas em *Softwares Livres*, porque seus funcionários não estão habituados à utilizá-los ou ainda, não o utilizam por "*resistência à mudanças*"? E quantas destas empresas deixaram de customizar um determinado *Software Livre* para atender sua necessidade por falta de verba para o pagamento dos serviços de profissionais (e liberar o código dos trabalhos desenvolvidos), substituindo-os por cópias de *softwares* ilegais que atendem suas necessidades? Com podem ver, esses são alguns exemplos dos efeitos prejudiciais da prática da pirataria, conforme acima mencionado nesta breve classificação. Além disso, os *softwares* proprietários também sofre prejuízos à parte, onde a maior perda está na renumeração pelo seu desenvolvimento.

O fator custo é preponderante nas questões de pirataria, de modo geral.

O CONCEITO FAIR USE (USO JUSTO)

Quando se realiza uma cópia de algum produto proprietário que restringe esta prática, este ato é considerado ilegal; porém este ato somente se tornará "*realmente*" ilegal quando terceiros vierem à utilizar a cópia deste produto sem licença de uso. Porém, mesmo que o usuário faça o uso deste recurso, a cópia – não autorizada – de um produto proprietário não é considerado ilegal, desde que seja feita a utilização deste somente por ele ou aquele que obtenha a licença para uso (ou em uma única máquina, se for programas de computadores e assim a licença especificar).

Este é o fundamento básico do conceito *fair use*, ou seja, o *uso justo*, pois o usuário possui o direito de uso do produto e o faz de acordo com os termos de exclusividade à ele concebido.

Uma boa aplicação deste conceito é a realização da cópia de segurança de aplicativos e utilitários, a gravação em fita cassete de músicas de seus CDs prediletos para escutar no carro, entre outras finalidades, desde que não sejam utilizados por terceiros.

SOBRE OUTRAS LICENÇAS LIVRES

✓ <<http://www.gnu.org/philosophy/license-list.pt.html>>.

Além das tradicionais licenças do *Projeto GNU*, existem inúmeras outras licenças de caráter livre, porém nem todas são compatíveis aos pilares da filosofia do *Software Livre*. Para isto foi elaborada uma listagem das principais licenças e uma classificação simples – subdivididas em compatíveis e incompatíveis à ideologia –, decrevendo uma breve explicação sobre suas diferenças. Consultem a página oficial do *Projeto GNU* para obterem maiores informações à respeito.

CONCLUSÃO

No *Brasil*, infelizmente não existe uma cultura que respeite as leis e normas que regem a proteção de direitos autorais. Frequentemente nos deparamos com atos ilícitos, como as vendas (e compra) de *CDs* e *DVDs* piratas, ou a realização de cópias de obras licenciadas apenas para a utilização pessoal. Todos estes atos refletem o principal (e talvez único) interesse dos usuários: o de usufruir dos benefícios destas obras à um preço baixíssimo.

A base da ideologia do *Software Livre* não está na possibilidade de ter bons programas à baixo custo. O ideal defendido está na liberdade de uso das obras regidas pelos termos das licenças abertas. Mesmo apesar desta ideologia permitir a livre distribuição das obras licenciadas, ela também impõe algumas restrições para garantir à todos estes direitos, como a obrigatoriedade de concessão do código fonte e a impossibilidade de restringir sua cópia. &;-D



III. A LPI

INTRODUÇÃO

Hoje em dia no mercado de trabalho, para que os profissionais da área de *TI* possam obter boas oportunidades, eles deverão comprovar suas qualidades e conhecimentos técnicos através de diversas formas: experiência de trabalho, conhecimentos técnicos (testes), indicações e certificações são alguns dos requerimentos mais solicitados. E no caso daqueles que trabalham com sistemas *GNU/Linux*, já que, face à existência de poucos cursos técnicos, como estes profissionais se auto-habilitaram?

Felizmente há uma demanda crescente por estes profissionais, onde à cada dia, mais e mais instituições de ensino começam à oferecer cursos de especialização para esta área. Mais ainda assim não resolvem a questão de comprovação de conhecimentos técnicos para aqueles que lidam com o sistema à mais tempo ou não têm tempo e/ou condições financeiras de bancar um bom treinamento. Para estes profissionais (e muitos outros que virão à existir), temos a certificação *LPI – Linux Professional Institute*.

O QUE É?

✓ <http://www.lpi.org/pt_br/>.

A *LPI* é uma organização sem fins lucrativos que surgiu em 1999, sediada no *Canadá* e estabelecida pela *Comunidade GNU/Linux*, que visa desenvolver um sistema de certificação para os sistemas *GNU/Linux* desvinculada à qualquer empresa, sistema, entidade ou órgão.

Dentre suas prioridades, está o estabelecimento de padrões de certificação independente, onde não há influências de fabricantes e distribuições. Com isto teremos aplicações de metodologias neutras para a certificação, que visam uma ampla aceitação geral por parte dos usuários e empresas.

John Maddog é o diretor executivo da *Linux Internacional*, além de vir presidir as provas aqui no *Brasil*. Já em seu país, ela é presidida por *Evan Leibovitch*, com mais de 20 anos de experiência em sistemas *Unix*.

PATROCINADORES

Mesmo sendo auto-suficiente, são inúmeros os órgãos e empresas que se aderiram à certificação *LPI* e deram apoio financeiro ao projeto. Desde empresas de grande porte com *IBM* e *HP* à tradicionais distribuições, como a *SuSe*, *Mandriva* e *TurboLinux*.

A verba arrecadada pelo pagamento dos requerimentos para a prova, além de auxiliar no custeamento da implantação do projeto e aplicação das provas, são utilizada para o financiamento de outros projetos e iniciativas benéficas ao *Software Livre*.

RECONHECIMENTO

A *LPI* já recebeu prêmios importantes, de grandes entidades como a *Linux's Journal Editor's Choice Award*, em 2001/2002 e *CRN Magazine Annual Certification Survey*, em 2002.

A CERTIFICAÇÃO

MODALIDADES

Existem duas modalidades distintas, porém dependentes:

| Nível / Descrição / Certificação | | |
|---|-------------------------|--------------------|
| <i>Nível 1</i> | Básico. | <i>LPI 101/102</i> |
| <i>Nível 2</i> | Gerenciamento de redes. | <i>LPI 201/202</i> |

Cada prova tem duração de *90 minutos*. Lembrem-se de que para obter a certificação do *Nível 2*, deveremos já estar certificados pelo *Nível 1*.

VANTAGENS

Dentre as principais vantagens que a certificação *LPI* oferece, destaca-se:

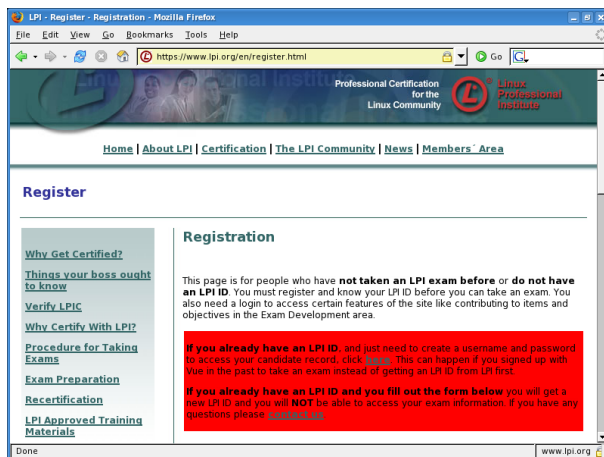
- A criação de uma certificação padrão universal;
- Inexistência de vínculo com empresas;
- Certificação sem “*data de validade*” ou expiração.

CADASTRO

- ✓ <<https://www.lpi.org/register.html>>.

Para realizar as provas do exame, será necessário efetuar o cadastro na página eletrônica oficial da organização.





Seção de registro da página oficial.

Ao final do processo, será fornecido um número especial, chamado *LPI ID*. Este é o registro (identificação), o qual será necessário para efetuar o pagamento da ficha de instrução, onde deverá constar escrito de forma legível esta numeração no recibo de depósito. Por último, deverá ser enviado via fax a ficha de instrução junto com o comprovante de pagamento. Nelas deverão estar anotados o nome e o *LPI ID*.

CALENDÁRIO

✓ <<http://www.lpibrasil.com.br/>>.

Periódicamente a *LPI* divulga na página oficial brasileira um calendário com os meses, as localidades e os centros de treinamento onde estarão sendo aplicadas as provas.



Para ter acesso à estas informações atualizadas, acessem a página oficial.

CONCLUSÃO

A comprovação de conhecimento por meio de certificações e diplomas é de grande valia para o sucesso profissional. Com certeza não seria diferente em se tratando de sistemas *GNU/Linux*, ainda mais sabendo que a grande maioria dos usuários sequer possuem (ou tiveram oportunidade de possuir) um treinamento técnico, face a poucas disponibilidades de instituições de ensino que primam ministrar cursos para estas qualificações técnicas. Enfim, para aqueles que desejam ingressar no mercado de trabalho com ênfase em utilização de sistemas *GNU/Linux*, é um caso a pensar! &;-D



IV. TEXTOS INTERESSANTES

INTRODUÇÃO

São inúmeros os textos que encontraremos na *Internet* caso desejarmos conhecer mais à fundo os conceitos e ideologias do *Software Livre*. Colocá-los aqui na íntegra ou até mesmo trechos não seria muito interessante, tal quantidade e diversidade de informações. Porém iremos inserir os endereços das páginas eletrônicas daqueles que foram os pioneiros no *Software Livre*.

OS TEXTOS...

PROJETO GNU

“Anúncio original do Projeto GNU”

- ✓ <<http://www.gnu.org/gnu/initial-announcement.pt.html>>.

Este atalho mostra o anúncio da existência do *Projeto GNU*, de construir um sistema operacional *Unix* completamente livre no ano de 1983.

“Combatendo patentes de software – uma a uma e todas juntas”

- ✓ <<http://www.gnu.org/philosophy/fighting-software-patents.pt.html>>.

Reflexões sobre os perigos que as patentes de *software* representam para o desenvolvimento de *Software Livre*, seus males e a forma de combatê-los.

“Palavras e frases confusas devem ser evitadas”

- ✓ <<http://www.gnu.org/philosophy/words-to-avoid.pt.html>>.

Algumas palavras e frases carregam certos conceitos que infelizmente não são aplicáveis ao *Software Livre*. *Stallman* descreve aquelas em que seu ponto de vista considera nociva ou prejudicial à disseminação da ideologia.

“Software Livre: Liberdade de cooperação”

- ✓ <<http://www.gnu.org/events/rms-nyu-2001-transcript.pt.txt>>.

Esta é uma transcrição de um discurso de *Richard Stallman* sobre *Software Livre* feita na *Universidade Nova York*, em 29 de maio de 2001.

“Nós podemos por um fim aos anexos (de e-mail) do Word”

- ✓ <<http://www.gnu.org/philosophy/no-word-attachments.pt.html>>.

Sobre os perigos e as armadilhas do uso do formato *.doc* nos anexos de mensagens de correio eletrônico. Em destaque, as artimanhas das empresas monopolistas e os prejuízos causados ao *Software Livre*.

FREE SOFTWARE FOUNDATION

“O que é Software Livre?”

- ✓ <<http://www.fsf.org/philosophy/free-sw.pt.html>>.

Definições claras da filosofia e concepção do *Software Livre*, onde sobressai a importância da liberdade (e não do preço).

“Porque o software não deveria ter donos”

- ✓ <<http://www.fsf.org/philosophy/why-free.pt.html>>.

Neste texto é ressaltada a importância da livre utilização da informação e tecnologia para o bem-comum da sociedade.

OUTROS

“A Cathedral e o Bazar”

- ✓ <<http://www.geocities.com/CollegePark/Union/3590/pt-cathedral-bazaar.html>>.

Escrito por *Eric S. Raymond*, este texto mostra as diferenças entre os modelos de desenvolvimento de *software* livre e comercial.

CONCLUSÃO

Liberdade! Este sempre foi o braço direito da filosofia duramente defendida pelos defensores do *Software Livre*! Estes textos esclarecem a grande importância do *Software Livre*, tendo a ética e a liberdade como fatores principais e preponderantes. &;-D



ENCERRAMENTO

O estudo, a interpretação e o entendimento de normas técnicas, padrões e legislação à primeira vista parece não ter muita importância ou utilidade para os mais novatos, vindo do lado prático. Mas por quê este assunto mereceu especialmente uma parte inteira dedicada neste livro?

Como quaisquer outro aplicativo, sistema operacional ou tecnologia, para os sistemas *GNU/Linux* e suas aplicações existe um conjunto de leis e normas que visam proteger a liberdade dos desenvolvedores e usuários, além de garantir a compatibilidade entre as diferentes idéias e concepções, e por fim a disseminação da ideologia do *Software Livre*.

O entendimento dos termos regidos nestes capítulos são de certa forma importantes, pois não só asseguram aos desenvolvedores, usuários e colaboradores do *Software Livre* como encorajam aos demais e as corporações à adotarem-no para atender às suas necessidades e exigências, despreocupando-os das possíveis limitações ou anomalias causadas pela falta de normas, padrões e legislações. &;-D

