Lojas Renner/CESAR School

Formação Encantech

Trilha de Engenharia de Dados

**Atividade 01**

Thiago Ney Evaristo Rodrigues

27 de abril de 2022

**Licenças de Software Livre**

**A Definição de Software Livre**

A definição de software livre apresenta os critérios utilizados para definir se um programa de computador em particular se qualifica como software livre ou não. De tempos em tempos nós revisamos essa definição, para esclarecê-la ou para resolver questões mais sutis.

**As quatro liberdades essenciais**

Um programa é software livre se os usuários possuem as quatro liberdades essenciais:

* A liberdade de executar o programa como você desejar, para qualquer propósito (liberdade 0).
* A liberdade de estudar como o programa funciona, e adaptá-lo às suas necessidades (liberdade 1). Para tanto, acesso ao código-fonte é um pré-requisito.
* A liberdade de redistribuir cópias de modo que você possa ajudar outros (liberdade 2).
* A liberdade de distribuir cópias de suas versões modificadas a outros (liberdade 3). Desta forma, você pode dar a toda comunidade a chance de beneficiar de suas mudanças. Para tanto, acesso ao código-fonte é um pré-requisito.

Um programa é software livre se ele dá aos usuários todas essas liberdades de forma adequada. Do contrário, ele é não livre. Enquanto nós podemos distinguir vários esquemas de distribuição não livres em termos de eles falham em serem livres, consideramos todos eles igualmente antiéticos.

**Software livre pode ser comercial**

“Software livre” não significa “não comercial”. Ao contrário, um programa livre deve estar disponível para uso comercial, desenvolvimento comercial e distribuição comercial. Essa política é de fundamental importância – sem ela, o software livre não poderia atingir seus objetivos.

**Software livre e licenciamento**

A popularização das licenças de software livre se deve ao fato da garantia jurídica que estas fornecem aos utilizadores de que não estão cometendo atos de infração de direito de autoria ao copiar ou modificar o software. Qualquer desenvolvedor pode estabelecer e criar suas próprias licenças de software. Entretanto a Open Source Initiative (OSI) recomenda fortemente que sejam reaproveitadas as licenças disponíveis e consolidadas pela comunidade de software livre. Tanto a OSI quanto o Projeto GNU listam as principais licenças de software livre existentes e suas descrições.

As licenças podem ser separadas, normalmente, em três categorias, divididas de acordo com a presença de termos que impõem restrições de licenciamento na redistribuição ou criação de trabalhos derivados do original. Desta forma as licenças são consideradas **permissivas** ou **recíprocas**. As recíprocas podem ser ainda consideradas **parciais** ou **totais**. A principal diferença é que as licenças recíprocas totais se caracterizam pela manutenção da licença original. As recíprocas parciais também podem ser conhecidas como *copyleft* fraco.

**Licenças Permissivas**

As licenças permissivas também podem ser conhecidas como licenças acadêmicas em referência às origens das licenças BSD da Universidade da Califórnia em Berkeley e MIT do Massachusetts Institute of Technology. Estas licenças recebem este nome pelas poucas restrições que impõem às pessoas que obtém o produto. Estas licenças devem ser usadas quando se deseja que o projeto possa atingir um número maior de pessoas para ampla divulgação. Um grande caso de sucesso neste caso é o servidor web Apache da Apache Software Foundation. A licença Apache, usada na maioria de seus produtos, é do tipo permissiva. Em geral, essas licenças permitem inclusive que, qualquer detentor do código possa desenvolver um produto derivado e fechar o código para explorá-lo comercialmente. Neste caso, o servidor web Apache se tornou um produto referência e tão amplamente usado que nenhuma empresa tem interesse em desenvolver um produto derivado com código fechado.

**Licenças Recíprocas Totais**

As licenças recíprocas totais têm a característica de que qualquer trabalho derivado do original deve ser redistribuído e disponibilizado sob os mesmos termos da licença original. Essa filosofia é conhecida como copyleft, termo criado pela Free Software Foundation. A ideia do copyleft é permitir a todos a execução, cópia, modificação e redistribuição das versões derivadas do software impedindo que possam ser acrescentadas restrições a essas modificações. Essa restrição determina que todo software livre produzido pela comunidade permaneça livre e não possa ter seu código fechado. A principal licença desta categoria é a GPL (GNU General Public Licence) da Free Software Foundation.

**Licenças Recíprocas Parciais**

As licenças recíprocas parciais também são conhecidas como copyleft fraco. Em princípio, determinam que modificações feitas em um software sob esta licença sejam disponibilizadas sob a mesma licença. Entretanto, se as modificações foram utilizadas como componente de outro projeto de software, este projeto não precisa, necessariamente, ser disponibilizado sob a mesma licença. Este é o principal diferencial das licenças recíprocas parciais para as recíprocas totais. Essas licenças possuem um equilíbrio entre as licenças permissivas e as recíprocas totais. Considera-se que as licenças recíprocas parciais proporcionam o interesse da comunidade de desenvolvedores pela liberdade que promovem. As licenças LGPL (GNU Lesser General Public Licence), MPL (Mozilla Public Licence) e EPL (Eclipse Public Licence) são exemplos desta categoria.

**Componentes do Sistema Operacional Linux**

O Linux é um sistema operacional baseado em Unix criado para desktops, mas que também é usado em servidores, smartphones, tablets e outros tipos de dispositivos, incluindo caixas bancários. Ao contrário de seus concorrentes mais famosos, o Linux não foi desenvolvido para fins comerciais e seu software e desenvolvimento são feitos em código aberto, o que significa que qualquer pessoa pode criar e distribuir aplicativos para ele.

A parte básica do Linux é composta de um kernel, camada criada para fazer a comunicação de outros programas e traduzi-los em comandos para a unidade de processamento e outros componentes eletrônicos.

A outra camada do sistema operacional é constituída por elementos que oferecem recursos capazes de garantir interação com o usuário. Popularmente referenciada como a camada de software. Esta camada permite que os aplicativos de usuário sejam executados. Entre outras palavras, o kernel do sistema não pode ser acessado diretamente pelo usuário ou administrador do sistema; isso só poderá ser possível através de aplicações utilitárias do sistema, bem como: terminal de linha de comando (CLI), softwares para compilação, software de gestão de disco/memória ou controle de processos do sistema, por exemplo.

**Estrutura do Linux**

Mesmo com todos os subsistemas do kernel (gerenciamento de processos, gerenciamento de memória, gerenciamento de redes, sistema de arquivos e outros), existem elementos que precisam compor essa base mantida no núcleo do sistema operacional para que ele funcione completamente. Entre eles destaco: o bootloader, o Shell, os softwares GNU, o servidor gráfico, o ambiente desktop e muito mais.

**Bootloader**

Um Bootloader (“carregador de inicialização”), também chamado de gerenciador de inicialização (boot), é um pequeno programa que carrega o sistema operacional de um computador na memória.

Quando um computador é ligado, a BIOS ou a UEFI realiza alguns testes iniciais das atividades básicas para o correto funcionamento de todos os recursos da máquina; e então transfere o controle para o Registro Mestre de Inicialização (MBR – Master Boot Record), onde o Bootloader se encontra.

Nos sistemas Linux, geralmente, o gerenciador de inicialização usado é o Grub. Com ele instalado, além de gerenciar o processo de inicialização do sistema, é possível ter vários sistemas operacionais instalados e escolher qual deve ser iniciado. O Grub fornece um menu que permite que você escolha entre as opções de sistemas disponíveis. Recurso comumente chamado de “dual-boot”.

Caso não possua diversos sistemas instalados na máquina (somente uma distribuição Linux), o Grub pode inicializar o sistema Linux quase que instantaneamente.

**Shell**

Primeiramente, o shell (ou interpretador de linha de comando) é um módulo que atua como camada externa (“concha”) entre o usuário e o sistema operacional. Existem diversos tipos de shell. O primeiro deles foi o Bourne shell (sh) que oferecia diversos comandos internos que permitiam ao usuário solicitar chamadas ao sistema operacional. A partir daí houveram evoluções significativas.

Atualmente, a maioria dos sistemas Linux usam, por padrão, uma evolução do Bourne shell, chamada Bash (Bourne Again Shell). O Bash, além das funcionalidades das versões anteriores, também implementa uma linguagem simples de programação que permite o desenvolvimento de pequenos programas (os famosos shell scripts).

Em resumo, o shell é um programa independente do usuário, executado fora do kernel, que fornece uma interface para interpretação de comandos. Ele permite a interação com o sistema executando comandos em uma interface de texto (CLI). Mesmo que você esteja apenas usando o ambiente gráfico e nunca tenha precisado usar ou executar nenhum comando Linux, o shell está em constante execução.

**Softwares GNU**

O shell fornece alguns comandos básicos embutidos, mas a maioria dos comandos que podem ser executados no shell Linux não são oferecidos por ele. Por exemplo, o comando cp (para copiar um arquivo), o comando ls (para listar os arquivos em um diretório) e comando rm (para apagar arquivos) são parte do pacote utilitários básicos GNU (“coreutils”). Nem todos os utilitários e programas de linha de comando são desenvolvidos pelo projeto GNU. Alguns comandos e programas de terminais possuem o seu próprio projeto independente.

Além deles, existem softwares utilitários essenciais para o funcionamento do sistema, como os utilitários de compilação e bootloader GRUB; também desenvolvidos e mantidos pelo projeto GNU. Assim, os sistemas Linux não iriam funcionar sem esses utilitários tão importantes.

**Servidor Gráfico**

O modo gráfico no Linux é gerado pelo servidor gráfico X (X Window System), que não é parte do kernel Linux. Entre outras funções, ele é responsável pela ativação da placa de vídeo, mouse e teclado; permitindo ao usuário o uso de interfaces gráficas que são chamadas de Gerenciadores de Janelas e Ambientes Desktops. Os Ambientes Desktops dispõem de interface completa para o usuário (GUI), bem como: barra de ferramentas, botões, ícones, wallpapers e bibliotecas gráficas. Já os Gerenciadores de Janelas dão base para os Ambientes Desktops.