

Respostas dos Exercícios

Linux

Simplicidade ao Seu Alcance

Tales Araujo Mendonça
Bruno Gonçalves Araujo



editora
VIENA

2ª Edição
Bauru/SP
Editora Viena
2012

Capítulo 1

1.

Richard Stallman é o pioneiro do software livre e fundou o projeto GNU que tem o objetivo de criar um sistema operacional completo, inclusive com todos os aplicativos necessários e que seja compatível com o sistema UNIX.

2.

Linus Torvalds resolveu criar um kernel para fins de estudo, e resolveu compartilhar o código com mais pessoas para conseguir ajuda no desenvolvimento do projeto. O nome Linux foi dado por Ari Lemmke , que colocou esse nome na pasta do servidor que compartilhava os arquivos do kernel que Linus estava desenvolvendo.

3.

O projeto GNU disponibilizava as ferramentas necessárias para se ter um ambiente completo, compatível com UNIX e feito somente com software livre, porém não possuía um kernel. Portanto a união das ferramentas GNU e o Kernel Linux possibilitou a criação de um sistema operacional completo e funcional.

Capítulo 2

1.

O software é a parte lógica, ou seja, o sistema operacional e todos os programas. Hardware é a parte física, os componentes como placa mãe, monitor, disco rígido.

2.

Todos os outros softwares funcionam a partir do sistema operacional. É o sistema operacional que se comunica diretamente com o hardware e gerência a forma como os programas terão acesso ao mesmo.

3.

- Ser um sistema multitarefas, capaz de executar várias tarefas simultaneamente.
- Ser um sistema multiusuário, permitindo que vários usuários utilizem o sistema ao mesmo tempo, com controle individual das ações de cada um.
- Seguir a licença GPL (Licença Pública Geral), que garante as 4 liberdades fundamentais do software livre: executar, estudar, redistribuir e aperfeiçoar o programa.
- Fazer parte do projeto GNU, que visava criar um sistema operacional totalmente livre e compatível com Unix, unindo-se ao kernel desenvolvido por Linus Torvalds.

4.

Liberdade nº. 0: A liberdade de executar o programa, para qualquer propósito; Liberdade nº. 1: A liberdade de estudar como o programa funciona e adaptá-lo para as suas necessidades. Acesso ao código fonte é um pré-requisito para esta liberdade; Liberdade nº. 2: A liberdade de redistribuir cópias, de modo que você possa ajudar ao seu próximo; Liberdade nº. 3: A liberdade de aperfeiçoar o programa e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie. Acesso ao código fonte é um pré-requisito para esta liberdade.

5.

Desenvolver aplicativos livres possibilita a interação de muitos colaboradores e cria a possibilidade do software se desenvolver mais rápido e melhor, pois terá mais “cabeças” pensando e contribuindo para atingir um objetivo comum. Assim como no software livre, sua filosofia de liberdade, colaboração e compartilhamento, pode ser aplicado na sociedade, onde apresenta uma situação de uns ajudando os outros para atingir um objetivo em comum.

6.

O Linux não possui uma cara, isso porque pode ser moldado e configurado de acordo com suas necessidades. A existência de vários ambientes gráficos, torna o Linux um sistema mais dinâmico e mutável que possa existir.

7.

LXDE e XFCE.

8.

Padrões abertos são descrições detalhadas e sem restrições de uso que informam as regras para ler e gravar um determinado tipo de arquivo. Interoperabilidade é quando um software pode trabalhar em conjunto com outro. Por exemplo, quando criamos um documento de texto em um programa e continuamos sua edição em outro. Para uma boa interoperabilidade é fundamental o uso de padrões abertos, pois assim diferentes programas terão como acessar e editar os mesmos arquivos.

9.

Software legado é quando um programa por algum motivo parou de receber atualizações, mas permanece sendo útil para resolver determinado problema e o trabalho de migração para programas mais recentes inviabiliza esse procedimento, dessa forma é mantido o programa desatualizado. Com o uso de software livre é possível que programas descontinuados passem a ser mantidos por novas equipes interessadas, bem como permite que sejam feitas melhorias pela equipe da empresa que utiliza o software.

Capítulo 3

1.

Distribuição ou distro, é um conjunto de pacotes, programas – incluindo o kernel – que juntas formam um sistema operacional Linux. As distribuições Linux são empacotadas e configuradas de acordo com cada empresa, equipe ou pessoa que a mentem.

2.

As distribuições Linux são formadas a partir de pequenos pacotes de software que podem ser adicionados ou removidos, determinando quais programas e recursos estarão disponíveis.

Além disso, as distribuições utilizam repositórios, que são servidores onde esses pacotes podem ser baixados e instalados de forma simples e geralmente automatizada no sistema Linux. Um repositório com uma boa quantidade de pacotes permite montar o sistema de acordo com as necessidades do usuário.

3.

Com o objetivo de compartilhar conhecimento e ter a oportunidade de fazer algo específico para cada um, começaram a surgir dezenas, milhares de distros Linux, cada uma com algo específico e próprio. Se uma distro não atendia determinado grupo, então surgiam novos desenvolvedores para sanar essa lacuna e criar sua própria distro.

4.

Não existe uma melhor distribuição Linux. Possuem algumas que são mais usadas e mais conceituadas, mas não quer dizer que sejam melhores. A melhor é a que mais te agrada e que te atende melhor a suas necessidades.

Capítulo 4

1.

Particionar, como a própria palavra sugere, significa dividir em partes ou separar em partes. Então, particionar o disco é justamente dividir o disco em partes. O particionamento de disco é utilizado, principalmente, para dividir e organizar o disco em partes. Dessa forma, torna-se possível, por exemplo, a instalação de vários sistemas operacionais no mesmo disco, cada sistema utilizando uma partição.

Capítulo 7

1.

Diretório é como uma pasta, um lugar para guardar, armazenar, arquivos e outros diretórios.

2.

Estrutura do diretório mari: /home/mari e Estrutura do diretório pendrive: /media/pendrives

3.

Diretório essenciais do sistema com direito de permissão apenas para administradores são encontrados em: /sbin e /usr/sbin

4.

Todos os arquivos de configuração do sistema, geralmente ficam concentrados no diretório /etc.

5.

Diretório /tmp que é usado para realizar tarefas que estão sendo executadas no sistema do usuário e são modificadas constantemente enquanto usuário estiver ativo no sistema.

6.

O diretório /var armazena arquivos que são modificados com frequência, como arquivos de logs que são muito úteis para analisar o que está a ocorrer no sistema ou em algum aplicativo específico.

Capítulo 8

1.

Sistemas de arquivos são regras para salvar e ler conteúdos salvos em mídias, por exemplo, discos rígidos e pendrives. Cada sistema de arquivos tem suas peculiaridades e tende a proporcionar vantagens para um determinado tipo de uso, entre os mais populares podemos citar: ext4, fat32 e xfs.

2.

O sistema de arquivos FAT é bastante antigo, sua primeira versão é de 1977 e foi utilizado como padrão nos sistemas DOS e Windows por muitos anos. Hoje, apesar do sistema de arquivos padrão no Windows ser NTFS o mesmo mantém suporte ao sistema FAT, assim como o sistema é suportado atualmente por praticamente todos os sistemas operacionais. Dessa forma, mesmo o sistema FAT estando bastante obsoleto, por questões de compatibilidade, continua sendo muito utilizado principalmente em pendrives e cartões de memória de câmeras e celulares.

Capítulo 9

1.
Execute o comando seguido do caractere &

2.

yasmin@ra:/tmp#

yasmin É o nome de usuário (login) da máquina que está logado.

@ (arroba) na informática significa at, do inglês, em. O usuário está na máquina.

o símbolo cerquilha (#) é de usuário com poderes de superusuário (root).

3.

Sim, é possível. Por padrão, em ambientes Linux, é utilizado o modo “case sensitive”, isso significa que para o sistema letras maiúsculas e minúsculas são diferentes.

4.

Pode-se executar com pontoBarra+nome-do-script (./hu_reconectar.sh); ou ainda utilizar a palavra bash<ESPAÇO>nome-do-script (bash hu_reconectar.sh).

5.

Tecla TAB

6.

Utilizando “sudo comando” teremos o comando sendo executado com permissões administrativas, ou seja, pelo usuário root. Já o “sudo su comando” irá apenas gerar um erro.

7.

ls -a

8.

Para voltar, anteceder um diretório, basta digitar (cd pontoPonto) “cd ..” Já para voltar ao diretório principal do usuário, basta digitar cd.

9.

Exibe qual diretório estamos acessando.

10.

Utiliza a opção -r, exemplo: cp -r

11.

mkdir -p Filmes/DIVX/Clássicos

12.

O comando ln cria atalhos, dessa forma é possível, por exemplo, criar um link no Desktop para o arquivo que está no endereço /usr/bin/firefox. A forma mais usual de utilizar o comando ln é com o parâmetro -s. Exemplo de uso: ln -s “/usr/bin/firefox” “\$HOME/Desktop/Firefox”

13.

Utilizar o comando passwd

14.

Para alterar somente a hora do sistema de forma simples, você pode utilizar o seguinte comando:

sudo date -s “HH:MM”

Onde:

sudo: é usado para executar o comando como superusuário (necessário para alterar data/hora).

date -s: indica que você quer definir (set) uma nova data/hora.

“HH:MM”: é onde você coloca a nova hora no formato horas:minutos entre aspas duplas.

Por exemplo, para definir a hora como 15:30 (3:30pm), você executaria:
`sudo date -s "15:30"`

Lembre-se de substituir “HH:MM” pelos valores de hora e minutos desejados. Esse comando altera apenas a hora, mantendo a data atual.

15.

Para fechar uma sessão utiliza-se o comando `exit`

16.

Executar comando `clear`

17.

Mover é quando se altera o diretório onde a pasta ou arquivo está contido, por exemplo mover um arquivo do diretório `/home/hu/musicas` para `/home/hu/videos`, enquanto renomear executa um procedimento similar, porém mantém o arquivo na mesma pasta e altera apenas o seu nome.

18.

Executar o comando `users`

19.

Para configurar uma rede cabeada (com fio) no Linux de maneira simples, você pode utilizar o comando `ifconfig` da seguinte forma:

`sudo ifconfig nome_da_interface_de_rede ip_desejado netmask mascara_de_rede`

Por exemplo, para configurar o endereço IP 192.168.1.100 com máscara 255.255.255.0 na interface `eth0`:

`sudo ifconfig eth0 192.168.1.100 netmask 255.255.255.0`

20.

Pode-se utilizar o comando `ping` pingando algum site externo a rede e se responder é que está na internet. Como por exemplo: `ping www.google.com`

21.

O comando `rm -rf` deve ser utilizado com cautela pois irá remover todos os arquivos e diretórios que estiverem contidos no endereço informado sem sequer pedir confirmação para efetuar a remoção. Um exemplo de algo muito perigoso seria `rm -rf /home` que removeria o diretório de todos os usuários.

22.

Para verificar o espaço total que os arquivos estão ocupando no pendrive e o que há de espaço livre:
`df -h /media/dados`

Para verificar apenas o espaço que os arquivos estão ocupando no pendrive:
`du -sh /media/dados`

23.

O comando `su` serve para efetuar login com algum usuário. Exemplo: `su tales` ou `su bruno`

24.

Utiliza-se o comando `free -m` para verificar quanto de memória RAM está sendo utilizada pelo sistema.

25.

Pode-se descobrir qual arquitetura do sistema utilizado o comando `uname -m`

26.

Compactar com `tar.gz`:
`tar -cvzf fotos.tar.bz2 /tmp/fotos`

Descompactar um `tar.gz`:
`tar -xvzf fotos.tar.bz2 -C ~/Imagens/fotos`

Compactar com `zip`:
`zip -r fotos.zip /tmp/fotos`

Descompactar um `zip`:
`unzip fotos.zip -d ~/Imagens/fotos`

27.

Para tornar o arquivo executável: `chmod +x /tmp/imposto.run`

Para mudar de usuário e grupo: `chown bruna:users /tmp/imposto.run`

28.

Basta utilizar a tecla de “seta” para cima no teclado.

29.

Para instalar ou remover pacotes utilizando uma única linha de comando, você pode encadear comandos com o operador `&&`, que executa o próximo comando apenas se o anterior for bem-sucedido. Aqui está o exemplo de como atualizar o sistema, instalar o pacote `k3b` e como remover o pacote `emesene` usando essa técnica:

```
sudo apt-get update && sudo apt-get upgrade && sudo apt-get install k3b &&  
sudo apt-get remove emesene
```

Capítulo 9

1.

Wine (acrônimo para “Wine Is Not an Emulator”, que em português significa “Wine não é um emulador”) é um projeto de software livre que permite a execução de programas desenvolvidos para o sistema operacional Windows em plataformas UNIX-like, como Linux, macOS e BSD. A principal finalidade do Wine é proporcionar compatibilidade para executar aplicativos Windows nesses sistemas operacionais, funcionando como uma camada de tradução para chamadas do sistema Windows para o Linux.

2.

Quando necessitar de uma função não encontrada em algum programa nativo para Linux, mas que está disponível em um programa para Windows, deve se utilizar o Wine para executar o programa diretamente no Linux. O uso do Wine não é sempre viável, pois trata-se de um projeto baseado em engenharia reversa que apesar de todos os esforços, é compatível com uma quantidade limitada de programas.

3.

Um: O CrossOver, que fez alterações no código fonte do Wine para melhorar o suporte a aplicativos e jogos populares, porém seu uso é pago.

Dois: O WineTricks que efetua a instalação e configuração de programas e jogos no Wine de forma automática, esse aplicativo é gratuito, porém a lista de programas e jogos disponíveis é pequena.

4.

O Play On Linux automatiza instalações de aplicativos e jogos no Wine. Em muitos casos é necessário instalar dependências e efetuar configurações específicas para um programa funcionar corretamente no Wine e o Play On Linux efetua automaticamente essas operações. O GameTree é uma variação do Wine que possui alterações em seu código fonte que o otimizam para executar jogos, o mesmo também conta com uma interface para gerenciar os jogos instalados.