

Minicurso Linux

Ananda Nunes, Filipe Cruz, Jorge Hermes, Matheus
William

Material de estudos para o minicurso de Linux que irá ser ministrado durante a semana acadêmica da UEPA.

.

Sumário

1. Fundamentos de Linux	5
1.1. O que é o Linux	5
1.2. Distribuições Linux	5
1.2.1. Debian	5
1.2.2. Fedora	5
1.2.3. Arch Linux	5
2. Estrutura de Arquivos no Linux (FHS)	6
2.1. O que é um sistema de arquivos	6
2.2. O que é um diretório	6
2.2.1. Principais diretórios do Linux	6
2.2.1.1. /home	6
2.2.1.2. /usr	6
2.2.1.3. /etc	6
2.2.1.4. /root	6
2.2.1.5. /opt	7
2.2.1.6. /lib	7
2.2.1.7. /boot	7
2.2.1.8. /sbin	7
2.2.1.9. /bin	7
2.2.1.10. /var	7
2.2.1.11. /mnt	7
2.2.1.12. /media	7
2.2.1.13. /tmp	7
2.2.1.14. /dev	7
2.2.1.15. /srv	7
2.2.1.16. /proc	8
2.3. Tipos de sistemas de arquivos	8
3. Editores de Texto de Linha de Comando	9
3.1. Vi	9
3.2. Vim	9
3.3. Nano	9
4. Comandos Básicos	10
4.1. Man	10
4.2. Comandos para manipulação de diretório	10
4.3. Comandos para manipulação de arquivos	12
5. Permissões de Arquivos no Linux	17
5.1. Exemplo de saída do arquivo (comando ls -l)	18
5.2. Comandos para definição de permissões	18
6. Instalação de Pacotes - Via Instalador no Terminal (Apt Install) e Arquivos (.deb, tar.gz e outros)	19
6.1. Função geral da instalação de pacotes:	19
6.2. Instalação via terminal	19
6.3. Instalação por arquivos	19

7. Caça ao Pinguin!	20
8. Desafio: Instalação da distro Arch Linux	21
9. Referências	22

1. Fundamentos de Linux

1.1. O que é o Linux

O Linux é um sistema operacional criado por Linus Torvalds, em 1991. Seu código fonte é liberado como Free Software (software livre), sob licença GPL.

Isto quer dizer que você não precisa pagar nada para usar o Linux, e não é crime fazer cópias para instalar em outros computadores, o que é um dos motivos da estabilidade e velocidade em que novos recursos são adicionados ao sistema.

No Linux, o Kernel Linux mais o conjunto de ferramentas GNU, compõem o sistema operacional Linux. Por isso é comum ler o termo GNU/Linux, ao se referir ao S.O.

1.2. Distribuições Linux

Distros de Linux são um pacote composto do GNU/Linux + Outras aplicações que compõem um sistema operacional. Geralmente, cada distro busca atender um público-alvo e direciona suas funcionalidades para satisfazer as necessidades desse público, seguindo uma filosofia de desenvolvimento.

1.2.1. Debian

Uma das distros mais antigas (desde 1993), o Debian é focado em estabilidade e por isso ele se torna uma das melhores escolhas para servidores, sendo utilizável por anos sem precisar de uma atualização.

1.2.2. Fedora

Uma distro patrocinado pela Red Hat, funciona como um campo de testes para tecnologias que serão implementadas no Red Hat Enterprise Linux (RHEL). Atende desenvolvedores que buscam estabilidade, mas sem sacrificar atualizações mais frequentes.

1.2.3. Arch Linux

O Arch segue a filosofia *KISS* (Keep It Simple, Stupid). Simples, nesse caso, significa que o sistema não vem com nada instalando por padrão, o usuário deve escolher quais pacotes instalar; incluindo o Kernel Linux e o Desktop Environment.

2. Estrutura de Arquivos no Linux (FHS)

2.1. O que é um sistema de arquivos

Um sistema de arquivos é uma estrutura usada por um sistema operacional para organizar e gerenciar arquivos em um dispositivo de armazenamento, como um disco rígido, SSD ou pendrive. Ele define como os dados são armazenados, acessados e organizados no dispositivo de armazenamento.

2.2. O que é um diretório

Um diretório é uma pasta, ou seja, é um local dentro do sistema de arquivos usado para guardar e organizar arquivos. Ele também pode conter outros diretórios, ajudando a manter tudo organizado em diferentes níveis.

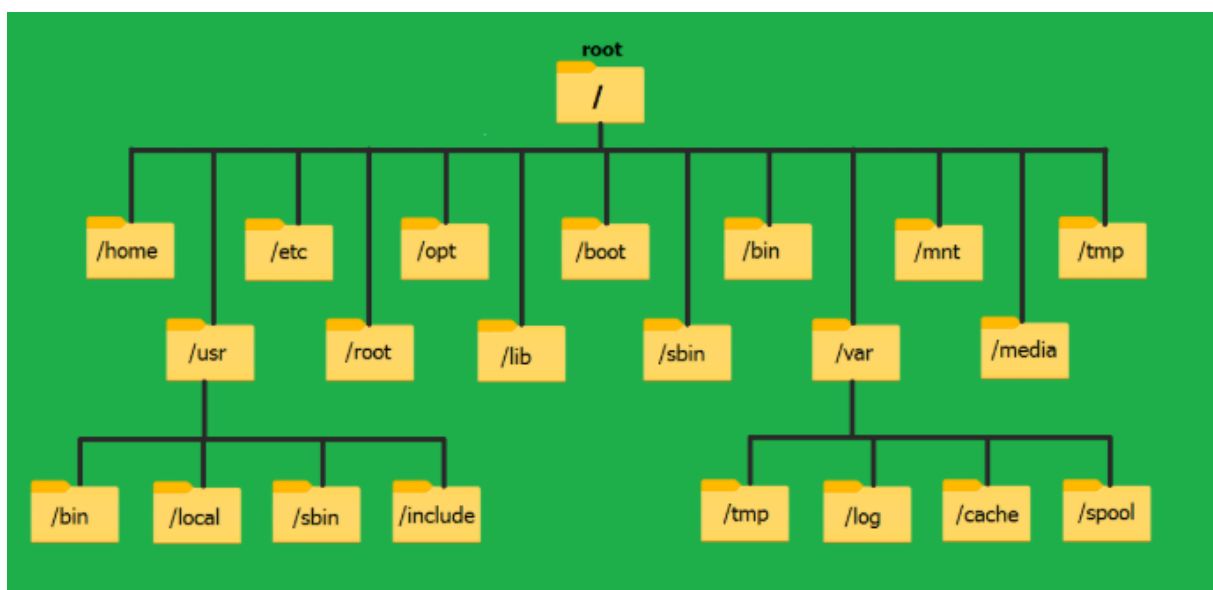


Figura 1: Filesystem Hierachy Standard - Estrutura de Hierarquia

2.2.1. Principais diretórios do Linux

2.2.1.1. /home

Diretório pessoal dos usuários. Cada usuário possui seu próprio subdiretório dentro desta pasta para armazenar seus arquivos pessoais.

2.2.1.2. /usr

Contém os diretórios e arquivos de uso comum do sistema, como programas, bibliotecas, documentação, entre outros. É uma das áreas mais extensas do sistema de arquivos.

2.2.1.3. /etc

Armazena arquivos de configuração do sistema, incluindo arquivos de inicialização (init.d) e configurações globais. É um diretório importante para administradores de sistema.

2.2.1.4. /root

Diretório pessoal do usuário root (superusuário). O superusuário é o administrador do sistema.

2.2.1.5. /opt

Destinado à instalação de pacotes de software adicionais de terceiros (opcionais).

2.2.1.6. /lib

Contém bibliotecas de código compartilhado usadas pelos programas do sistema.

2.2.1.7. /boot

Armazena arquivos relacionados ao processo de inicialização (boot) do sistema, como o bootloader e o kernel.

2.2.1.8. /sbin

Contém utilitários do sistema para a administração e operação do sistema. Esses utilitários são armazenados em arquivos relacionados ao processo de inicialização (boot) do sistema, como o bootloader e o kernel. geralmente acessíveis apenas pelo superusuário.

2.2.1.9. /bin

Contém os utilitários (programas) essenciais para o funcionamento básico do sistema, acessíveis a todos os usuários.

2.2.1.10. /var

Armazena arquivos de dados variáveis, como arquivos de log, banco de dados, spool de impressão, entre outros. Esses arquivos são alterados regularmente durante a operação do sistema.

2.2.1.11. /mnt

Diretório para montagem temporária de sistemas de arquivos adicionais.

2.2.1.12. /media

Diretório de montagem automática para dispositivos de mídia removíveis, como CDs, DVDs e dispositivos USB.

2.2.1.13. /tmp

Diretório para arquivos temporários usados pelos programas em execução no sistema. Os arquivos armazenados neste diretório

2.2.1.14. /dev

Contém os dispositivos do sistema, como discos rígidos, unidades USB, impressoras, entre outros. Cada dispositivo é representado por um arquivo especial.

2.2.1.15. /srv

Diretório que contém dados específicos do serviço, como arquivos de log, arquivos de configuração de sites, entre outros.

2.2.1.16. /proc

Sistema de arquivos virtual que fornece informações sobre os processos em execução e o estado do sistema.

2.3. Tipos de sistemas de arquivos

SISTEMA OPERACIONAL	SISTEMAS DE ARQUIVOS SUPORTADOS
Linux	EXT3, EXT4, BTRFS, XFS, JFS
MacOS	HFS
Windows	FAT, HPFS, NTFS

Tabela 1: Tipos de Sistemas de Arquivos por S.O.

- EXT3: usado como padrão no Linux a partir de 2001. Possui journal, que ajuda a recuperar o sistema em caso de desligamento inesperado. Suporta até 16 TB no sistema de arquivos, 2 TB por arquivo e até 32 mil subpastas em uma pasta.
- EXT4: padrão do Linux desde 2008. É mais rápido e moderno que o EXT3. Suporta até 1 EB no sistema de arquivos, 16 TB por arquivo e ilimitadas subpastas em uma pasta.

3. Editores de Texto de Linha de Comando

Programas que permitem a criação e edição de arquivos pela linha de comando.

3.1. Vi

Editor texto padrão de muitos sistemas POSIX. Ele utiliza de diferentes “modos” de edição para permitir mais funcionalidades.

Comandos úteis:

- Modo NORMAL (Aperte <ESC> para entrar nesse modo)
 - utilize <h, j, k, l> para mover o cursor
 - :q! -> Sair sem salvar
 - :w -> Salvar
 - :wq -> Sair e Salvar
- Modo INSERÇÃO/INSERT (Aperte <i> para entrar nesse modo)
 - Esse modo permite a inserção de texto.

3.2. Vim

Um sucessor do Vi, adiciona várias opções de qualidade de vida como o realce de sintaxe, plugins e o modo visual. Utiliza a maioria dos comandos do Vi.

3.3. Nano

Editor de texto simples e de fácil uso. Não tem modos como o Vi ou Vim.

Comandos úteis:

- Salvar: CTRL + O
- Sair: CTRL + X

4. Comandos Básicos

4.1. Man

As páginas de manual dos comandos são a principal fonte de documentação do Linux quando você tem alguma dúvida sobre um comando. Essas páginas podem ser acessadas da seguinte forma:

```
man [nome do comando]
```

As páginas man oferecem um breve resumo da finalidade, da sintaxe e das opções associadas a um comando específico.

A documentação exibida de um comando normalmente terá muitas páginas. Para navegar pelas páginas, use os botões a seguir do teclado:

- Tecla de **seta** para **cima** ou para **baixo**: rola para cima ou para baixo uma linha, respectivamente
- **Page Up** ou **Page Down**: rola uma página para cima ou para baixo, respectivamente
- **Barra de espaço**: rola uma página para baixo

Você também pode pesquisar a página man de um comando usando o caractere de barra (/):

```
/<searchString>
```

Para sair das páginas de manual, digite **q**.

4.2. Comandos para manipulação de diretório

- **ls**

Lista os arquivos de um diretório.

Cores diferentes representam diferentes tipos de arquivos

```
ls [opções][caminho/arquivo][caminho1/arquivo1]...
```

OPÇÃO	DESCRIÇÃO
-l	Formato longo (mostra permissões)
-h	Tamanhos de arquivo relatados em um formato amigável para humanos
-s	Mostra o tamanho de cada arquivo e o tamanho do diretório
-a	Mostra todos os arquivos, incluindo arquivos ocultos
-R	Lista subdiretórios
-X	Classifica em ordem alfabética por extensão de arquivo
-S	Classifica por tamanho de arquivo
-t	Classifica por tempo de modificação

Tabela 2: Opções úteis para o comando ls

- cd

Entra em um diretório no qual você tem permissão de execução.

cd [*diretório*]

- mkdir

Cria um diretório no sistema

mkdir [opções][*caminho/diretório*][*caminho1/diretório1*]...

4.3. Comandos para manipulação de arquivos

- cat

Mostra o conteúdo de um arquivo binário ou texto

cat[opções][*diretório/arquivo*][*diretório1/arquivo1*]...

- rm

Apaga arquivos, diretórios e subdiretórios vazios o que contenham arquivos.

rm [opções][*caminho*][*arquivo/diretório*][*caminho1*][*arquivo1/diretório1*]...

Opção	Descrição
-d	Remove um diretório se o diretório estiver vazio
-r	Força a remoção de um diretório não vazio
-f	Não envia prompt ao usuário. Útil para diretórios com muitos arquivos
-i	Interativo - envia prompt para confirmar cada arquivo
-v	Exibe nomes dos arquivos excluídos

Tabela 3: Opções úteis para o comando rm

- cp

Copia arquivos e diretórios.

Por padrão, o comando sobrescreve arquivos existentes com o mesmo nome.

cp [opções][*origem*][*destino*]

OPÇÃO	DESCRIÇÃO
-a	Arquiva arquivos
-f	Força cópia sobrescrevendo o arquivo de destino, se necessário
-i	Interativo - Pergunta antes de sobrescrever
-n	Não sobrescreve arquivos
-v	Modo detalhado - Imprime mensagens informativas
-l	Vincula arquivos em vez de copiar

Tabela 4: Opções úteis para o comando cp

- mv

Move ou renomeia arquivos e diretórios. Executa o mesmo processo do cp, mas apaga o arquivo de origem no final do processo.

mv [opções][*origem*][*destino*]

OPÇÃO	DESCRIÇÃO
-i	Envia um prompt antes de sobrescrever um arquivo
-f	Evita receber prompts
-n	Não sobrescreve arquivos existentes
-v	Mostra o nome dos arquivos que são movidos ou renomeados

Tabela 5: Opções úteis para o comando mv

É possível utilizar uma expressão regular para mover arquivos do mesmo tipo.

mv *.png dir1

Move todos os arquivos do tipo **.png** para **dir1**

- find

Pesquisa arquivos que correspondem a critérios específicos em um diretório designado.

Pode pesquisar por:

- Proprietário
- Nome do Arquivo
- Tamanho do arquivo
- Data de modificação do arquivo
- Tipo de arquivo

find [diretório de origem][opções][o que encontrar]

OPÇÃO	DESCRIÇÃO
-name	Pesquisa por nome de arquivo
-iname	Pesquisa por nome de arquivo, mas ignora letras maiúsculas
-user	Pesquisa por proprietário do arquivo
-type	Pesquisa por tipo de arquivo

Tabela 6: Opções úteis para o comando find

- grep

Pesquisa o conteúdo de um arquivo em busca de um determinado padrão de texto ou string e exibe cada ocorrência.

grep [opções][texto a ser procurado][diretório onde procurar]

OPÇÃO	DESCRIÇÃO
-i	Ignora maiúsculas
-r	Executa pesquisas recursivas
-l	Lista apenas nomes de arquivos
-n	Exibe o número de linha
-c	Contagem de linhas de correspondentes

Tabela 7: Opções úteis para o comando grep

- tar

Compacta uma coleção de arquivos em um único arquivo para facilitar a cópia ou o download. O pacote criado é chamado de tarball

O conteúdo de um arquivo compactado pode, opcionalmente, ser compactado.

O comando também é usado para descompactar um arquivo

tar [opções][*nome_do_arquivo.tar*][*arquivo1/diretório1*][*arquivo2/diretório2...*]

Exemplos:

- Para compactar arquivos em um tarball:
tar -cvf tarball.tar file1 file2 file3
- Para compactar arquivos utilizando compressão:
tar -cafz tarball.tar.gz file1 file2 file3
- Para descompactar arquivos desse tarball:
tar -xaf tarball.tar
tar -xaf tarball.tar.gz (arquivo com compressão)
- Para compactar o tarball utilizando gzip
gzip tarball.tar

OPÇÃO	DESCRIÇÃO
-c	Cria um novo tarball
-x	Extraí o conteúdo de um tarball
-z	Compacta o conteúdo de um tarball usando o utilitário gzip
-a	Detecta automaticamente o tipo de compressão necessário, ao ler a extensão do arquivo
-f	Especifica o nome do tarball
-v	Produz uma saída detalhada mostrando os nomes dos arquivos enquanto o tarball é processado

Tabela 8: Opções úteis para o comando tar

5. Permissões de Arquivos no Linux

No Linux, cada arquivo e pasta tem permissões que controlam quem pode ler (r), escrever (w) ou executar (x). Essas permissões são divididas em três grupos: dono (owner), grupo (group) e outros (others). Assim, o sistema garante que cada usuário tenha apenas o acesso necessário.

	UTILIZADORES
u	User (Owner)
g	Grupo (Group)
o	Outros (Other)
a	All (Todos)

Tabela 9: Tabela Auxiliar 1

	PERMISSÕES
r	Leitura (Read)
w	Escrita (Write)
x	Executar (Execute)

Tabela 10: Tabela Auxiliar 2

	OPERADORES
+	Adiciona permissão
-	Remove permissão
=	Define permissão, remove as restantes (para u, g, o, a)

Tabela 11: Tabela Auxiliar 3

5.1. Exemplo de saída do arquivo (comando ls -l)

Como Ler Permissões de Arquivos

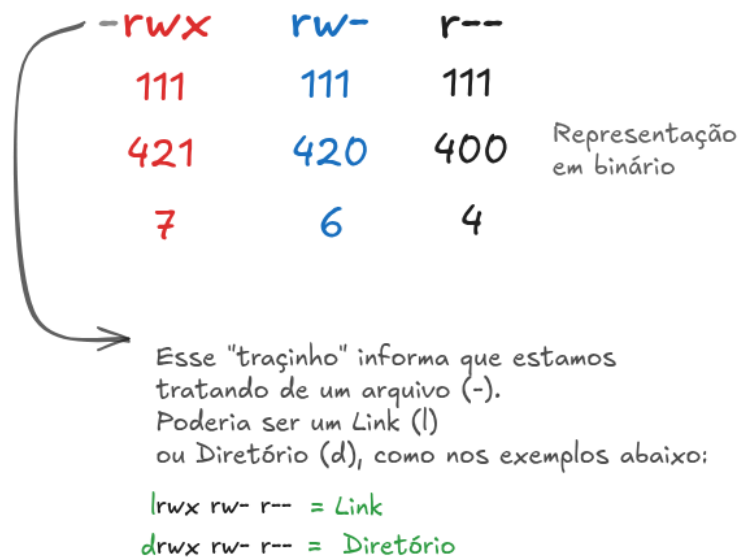


Figura 2: Como ler permissões de arquivos

Exemplo: `-rw-rw-r-- 1 aluno curso 1024 Feb 13 10:55 teste`

`-`: tipo do arquivo (`-` = arquivo comum, `d` = diretório, `l` = link, etc.)

`rw-`: permissões do dono (read, write, sem execução)

`rw-`: permissões do grupo (read, write, sem execução)

`r--`: permissões de outros (read, sem write, sem execução)

`1`: número de links ou hard links para o arquivo

`aluno`: dono do arquivo

`curso`: grupo do arquivo

5.2. Comandos para definição de permissões

- `chmod`: usado para alterar as permissões de leitura, escrita e execução de arquivos e pastas.
Exemplo: `chmod 700 arquivo` ou `chmod u+rwx arquivo`
- `chown`: usado para mudar o dono (usuário e grupo) de um arquivo ou diretório.
Exemplo: `chown usuario:grupo arquivo`

6. Instalação de Pacotes - Via Instalador no Terminal (Apt Install) e Arquivos (.deb, tar.gz e outros)

6.1. Função geral da instalação de pacotes:

O objetivo é adicionar novos softwares ou bibliotecas ao sistema de forma organizada, garantindo que todos os arquivos necessários sejam instalados, que dependências sejam resolvidas e que o programa funcione corretamente. Pacotes simplificam a manutenção do sistema e evitam conflitos, além de permitir atualizações fáceis.

6.2. Instalação via terminal

- apt update

Comando utilizado em distribuições Debian/Ubuntu para instalar pacotes diretamente dos repositórios oficiais. Ele baixa automaticamente os arquivos necessários, resolve dependências e instala o software de forma segura e padronizada. É a forma mais prática de manter o sistema atualizado e consistente.

Ex: `sudo apt update && sudo apt install vim`

- pacman -Syu

Comando utilizado em distribuições Arch Linux e derivadas para instalar e atualizar pacotes. -S instala pacotes, -y sincroniza a base de pacotes, e -u atualiza o sistema. Assim como o APT, ele resolve dependências automaticamente, mas é específico para sistemas baseados em Arch.

Ex: `sudo pacman -Syu && sudo pacman -S firefox`

6.3. Instalação por arquivos

- Arquivos .deb:

Pacotes pré-compilados para Debian/Ubuntu, contendo todos os arquivos e informações necessárias para instalação. São úteis quando o pacote não está disponível nos repositórios oficiais.

Ex: `sudo dpkg -i pacote.deb`

- Arquivos .tar.gz e outros compactados:

Arquivos que normalmente contêm o código-fonte do software. Para instalar, é necessário descompactar, configurar, compilar e instalar manualmente. Esse método oferece maior flexibilidade, mas exige conhecimento sobre dependências e configurações do sistema.

Ex: `tar -xzf pacote.tar.gz`

`cd pacote`

`./configure`

`make`

`sudo make install`

7. Caça ao Pinguin!

Dinâmica envolvendo os conhecimentos adquiridos sobre Linux até agora.

8. Desafio: Instalação da distro Arch Linux

Em sua casa, tente instalar a distribuição Arch Linux, mas sem usar o script *archinstall* !
Para mais informações refira-se ao guia de instalação presente na wiki oficial do Arch:

Arch Wiki em PT-BR °

Instalação da ISO do Arch °

Caso queira utilizar um ambiente isolado no seu computador, tente a **Oracle Virtual Box** °

9. Referências

- [Guia Foca Linux \(Iniciante + Intermediário\)](#)[◦]
- [Roadmap.sh \(Linux\)](#)[◦]
- [Permissões de acesso no Linux](#)[◦]
- [Instalação de tar.gz, tar.xz, tar.bz2](#)[◦]
- [Instalação de tar.gz no Linux](#)[◦]
- [Partições no Linux](#)[◦]
- [Estrutura de diretórios no Linux](#)[◦]
- [Entendendo a Hierarquia do Sistema de Arquivo do Linux](#)[◦]
- [Sistema de Arquivos no Linux](#)[◦]
- [Hierarquia de Sistema de Arquivos no GNU/Linux](#)[◦]
- [Arch Wiki](#)[◦]
- [Debian Wiki](#)[◦]
- [Fedora Wiki](#)[◦]