



# Manual GNU/Linux



## Tabela de conteúdos

Manual GNU/Linux	1.1
Introdução	1.2
O shell e o terminal	1.3
Gerenciamento de arquivos	1.4
Controle de permissões	1.5
Glossário	1.6
Bibliografia	1.7

# **Apresentação**

Esse manual é uma ferramenta de aprendizado institucional. Ele condensa o conhecimento acumulado a respeito do GNU/Linux e o disponibiliza para todos os membros da organização.

Para que o conteúdo esteja sempre atualizado, é importante que cada funcionário dedique parte do seu tempo para inserir seus aprendizados — relacionados ao tema GNU/Linux — neste manual. Dessa forma, é possível tornar toda descoberta local em aprendizado global.

## O que é GNU/Linux?

O GNU/Linux é um sistema operacional *Open Source* composto pelo *kernel* Linux, criado em 1991 por Linus Torvalds, e pelo conjunto de ferramentas GNU, lançado em 1983 por Richard Stallman.

## As distribuições GNU/Linux

No mundo todo existem grupos de pessoas, empresas e organizações que decidem usar o GNU/Linux como base para criar e distribuir, de diferentes formas, sistemas operacionais com outros *softwares*. Esses novos sistemas são chamados de distribuições GNU/Linux.

Existem distribuições que não nascem diretamente do GNU/Linux, mas sim de outras. Atualmente, as distribuições GNU/Linux mais populares nasceram a partir do Debian ou do Red Hat.

## O interpretador de comandos

Popularmente conhecido como *shell*, é o programa responsável por interpretar as instruções enviadas pelo usuário e seus programas ao *kernel*. Ele que executa comandos lidos do teclado ou de um arquivo executável. O GNU/Linux possui diversos tipos de interpretadores de comandos, entre eles o mais usado é o *bash*.

Os comandos podem ser enviados de duas maneiras para o *shell*, são elas:

- Interativa: Os comandos são digitados e passados ao interpretador de comandos um a um. Neste modo, o computador depende do usuário para executar uma tarefa, ou próximo comando.
- Não-interativa: São usados *scripts* para o computador executar os comandos na ordem encontrada no arquivo. Neste modo, o computador executa os comandos do arquivo um por um e dependendo do término do comando, o script pode checar qual será o próximo comando que será executado e dar continuidade ao processamento.

## Terminal virtual

Também conhecido como "console" ou simplesmente "terminal", é um programa que executa um *shell*. Com ele é possível abrir várias seções de trabalho completamente independentes umas das outras. Essas seções podem estar conectadas ao *shell* de um computador local ou remoto.

## Uso do terminal

Ao iniciar o terminal ele exibirá uma estrutura semelhante a esta:

```
usuario@maquina:local$
```

. Os caracteres antes do `@` são o nome do usuário que está operando a máquina. Após o `@` e antes do `:` é exibido o nome da máquina. Entre o `:` e o `$` é informado o local que está sendo acessado no sistema operacional. O `$` significa que o usuário logado não tem a credencial *root* — caso tivesse o símbolo exibido seria o `#`. Após esse conteúdo, é exibido um `|` ou um `-` ele indica que o terminal está apto a receber comandos e onde o texto inserido será colocado.

## Arquivos no GNU/Linux

Os arquivos são utilizados para gravar dados. Eles, podem conter um texto, uma música, programa, planilha, etc.

### Nomes e extensões

Para facilitar a identificação, cada arquivo deve ter um nome — lembrando que o GNU/Linux é *case sensitive*. No entanto, muitas vezes, não é possível localizar ou saber como utilizá-lo somente pelo nome, por conta disso, existem as extensões.

A extensão são os caracteres separados do nome do arquivo por um `.`, assim: `nome_arquivo.extensao`. O uso de extensões não é obrigatório na maioria das distribuições GNU/Linux, mas é conveniente o seu uso para determinar facilmente o tipo de arquivo e qual programa utilizar para abri-lo.

No GNU/Linux, quando o primeiro caractere do nome é um `.`, significa que o arquivo em questão é oculto. Arquivos ocultos são ignorados, ou seja, eles existem, mas não são listados — a não ser que seja explicitado.

### Tipos de arquivos

No GNU/Linux, existem sete tipos de arquivos, são eles:

- Arquivo regular: É o tipo mais comum de arquivo que se pode encontrar em um sistema GNU/Linux. Existem dois tipos de arquivos regulares:
  - Arquivo de texto: Seu conteúdo é compreendido pelas pessoas. Ele pode ser uma carta, um *script*, um programa de computador escrito pelo programador, arquivo de configuração, etc.
  - Arquivo binário: Seu conteúdo somente pode ser entendido por computadores. Ele é gerado por um arquivo de programa através de um processo chamado de compilação.
- Arquivo de diretório: Armazena arquivos de qualquer tipo, inclusive outros diretórios.
- Arquivo de dispositivo: São utilizados para gerenciar os dispositivos de entrada e saída. Eles se subdividem em dois grupos:
  - Arquivo de caractere: Nele, normalmente, os dados são lidos e escritos diretamente no dispositivo, dispensando o uso de *buffers*.
  - Arquivos de bloco: Nele, as operações de entrada e saída são realizadas de modo aleatório, em blocos, fazendo o uso de *buffers* intermediários.
- Arquivo de *socket*: Permite a comunicação entre dois processos que compartilham dados. Ele permite a comunicação até mesmo entre sistemas operacionais diferentes.
- Arquivo *pipe*: Permite a comunicação entre dois processos executados no mesmo sistema operacional.

- Arquivo de *link* simbólico: Armazena uma representação textual do caminho para um arquivo referenciado.

## Diretórios no GNU/Linux

O diretório é um arquivo utilizado para armazenar conjuntos de arquivos. Com os diretórios é possível manter os arquivos organizados.

No GNU/Linux não é permitido a existência de dois arquivos com o mesmo nome no mesmo diretório. As exceções são:

- Arquivos de mesmo nome com extensões diferentes.
- Dois arquivos com o mesmo nome, e até com a mesma extensão, com um deles oculto.

## Estrutura de diretórios do GNU/Linux

O sistema GNU/Linux possui uma estrutura básica de diretórios. Essa estrutura também é conhecida como Árvore de Diretórios, porque é parecida com uma árvore de cabeça para baixo. Cada diretório do sistema tem seus respectivos arquivos que são armazenados conforme regras definidas pela *Filesystem Hierarchy Standard (FHS)* da seguinte forma:

- `/` : Diretório principal do sistema. Dentro dele estão todos os diretórios do sistema.
- `/bin` : É reservado para gravar comandos que serão utilizados por todos os usuários.
- `/boot` : Contém os arquivos necessários para a inicialização do sistema.
- `/cdrom` : Este diretório não faz parte do padrão de hierarquia *FHS*, porém ele ainda é encontrado em algumas distribuições GNU/Linux. É utilizado como local temporário para *CDs* e *DVDs* inseridos no computador — porém, o local padrão para essas mídias é o diretório `/media`.
- `/dev` : Contém arquivos gerados pelos dispositivos de *hardware*, como processador, placa de vídeo, leitor de mídia etc.
- `/etc` : Guarda a maioria dos arquivos essenciais do sistema operacional do computador local, como configuração usados para controlar uma operação ou programa.
- `/home` : Local destinado para os arquivos dos usuários, com exceção do *root*.
- `/lib` : Contém diretórios e *links* simbólicos compartilhados pelos programas do sistema e módulos do *kernel*.
- `/lost+found` : Local para a gravação de arquivos corrompidos, recuperados após uma verificação do sistema de arquivos — esse tipo de verificação ocorre na primeira reinicialização após um travamento no sistema.
- `/media` : Ponto de montagem de dispositivos removíveis como *pendrives*, *CDs*, *Blu-Ray* etc.
- `/mnt` : Ponto de montagem de subdiretórios temporários de dispositivos não removíveis e arquivos, como imagens ISO.

- `/opt` : É uma área reservada para instalações de pacotes de aplicações que otimizam o funcionamento de determinado programa ou acrescentam novos recursos.
- `/proc` : É usado pelo *kernel* para guardados registros de desempenho e status de processos.
- `/root` : Diretório padrão do usuário *root*.
- `/run` : Fornece às aplicações um local para armazenamento de arquivos temporários que, diferentemente dos arquivos armazenados no `/tmp` , em caso de exclusão causam problemas às aplicações que os utilizam.
- `/sbin` : Reúne arquivos binários acionados pelo sistema em si ou pelo *root* em processos de manutenção.
- `/srv` : Possui dados que são utilizados por serviços armazenados, como *web servers*.
- `/sys` : Sistema de arquivos do *kernel*, que facilita a troca de informações entre os programas que rodam no espaço do *kernel*, como os *drivers*, com os programas que rodam no espaço do usuário — por isso, esse diretório é usado por diversos programas.
- `/tmp` : Diretório para armazenamento de arquivos temporários criados por programas.
- `/usr` : Contém a maior parte dos programas. Normalmente acessível somente como leitura.
- `/var` : Aqui são armazenados logs e arquivos variados que normalmente seriam escritos em `/usr` .

## Manipulação de arquivos

Abaixo estão listados comandos úteis para a manipulação de arquivos.

### ls

- Utilidade: Listar os arquivos de um diretório.
- Estrutura: `ls [opções] [caminhos|arquivos]` .
- Opções:
  - `-a` : Lista todos os arquivos, inclusive os ocultos, de um diretório.
  - `-l` : Lista permissões, donos, grupos, tamanho em *bytes* e data de modificação dos arquivos.
  - `-lh` : Edita o tamanho dos arquivos mostrando pela opção `-l` , ou seja, o tamanho dos arquivos serão exibidos em *Kbytes*, *Mbytes* ou *Gbytes* — o que o GNU/Linux julgar mais conveniente.
- Macetes:
  1. `ls -lha` é uma concatenação das opções `-l` , `-h` , `-a` . Quando se trata de listagem, talvez, seja o comando mais completo.
  2. É possível listar vários diretório com um único comando. Por exemplo, o comando `ls . /` retornará a listagem do diretório atual e do diretório raiz do GNU/Linux.

### cd



- Utilidade: Entrar em um diretório. O usuário precisa ter a permissão de execução para entrar no diretório.
- Estrutura: `cd [caminho|diretório]` .
- Macetes:
  1. `cd /` : Levará ao diretório raiz do GNU/Linux.
  2. `cd ~` : Levará para o diretório padrão do usuário que está operando o GNU/Linux.
  3. `cd -` : Levará ao último diretório acessado.
  4. `cd ..` : Levará à um diretório a cima.
  5. `cd ../ [caminho|diretório]` : Levará à um diretório a cima e entrará, caso exista, no diretório do caminho informado.

## pwd

- Utilidade: Mostrar o nome e caminho do diretório atual.

## mkdir

- Utilidade: Criar um diretório no GNU/Linux.
- Estrutura: `mkdir [opções] [caminhos|diretórios]` .
- Opções:
  - `-p` : Caso os diretórios dos níveis acima não existam, eles também serão criados.
  - `-v` : Mostra uma mensagem para cada diretório criado.
- Macetes:
  1. Para criar um novo diretório, o usuário deve ter permissão de gravação.
  2. É possível criar vários diretório com um único comando. Por exemplo, o comando `mkdir diretorio1 diretorio2` criará dois diretórios — o `diretorio1` e o `diretorio2` .

## rmdir

- Utilidade: Remover um diretório vazio do GNU/Linux.
- Estrutura: `rmdir [caminhos|diretórios]` .
- Macetes:
  1. Para remove um diretório, o usuário deve ter permissão de gravação sobre ele.
  2. É possível remover vários diretório com um único comando. Por exemplo, o comando `rm diretorio1 diretorio2` removerá tanto o `diretorio1` quanto o `diretorio2` — desde que ambos estejam vazios.
  3. Para remover um diretório que contém arquivos, basta usar o comando `rm` com a opção `-r` seguida pelo diretório que deseja remover.

## cat

- Utilidade: Mostrar o conteúdo de um arquivo binário ou de texto.
- Estrutura: `cat [opções] [caminhos|arquivos]` .

- Opções:
  - `-n` : Mostra o número das linhas enquanto o conteúdo do arquivo é mostrado.

## touch

- Utilidade: Criar um arquivo regular em branco
- Estrutura: `touch [caminhos|arquivos]` .
- Macetes:
  1. É possível criar vários arquivos com um único comando. Por exemplo, o comando `touch arquivo1 arquivo2` criará dois arquivos — O `arquivo1` e o `arquivo2` .
  2. Para escrever algo no arquivo basta utilizar um editor de texto como por exemplo o Nano ou o Vim.

## rm

- Utilidade: Remover arquivos.
- Estrutura: `rm [opções] [caminhos|diretórios]` .
- Opções:
  - `-i` : Pergunta antes de remover um arquivos.
  - `-v` : Mostra os arquivos a medida que são removidos.
  - `-r` : Remove arquivos em subdiretórios e em seguida o subdiretório.
- Macetes:
  1. `rm -r [caminho|diretório]` é utilizado para remover um diretório.

## cp

- Utilidade: Copiar arquivos.
- Estrutura: `cp [opções] [origem] [destino]`
- Opções:
  - `-r` : Copia arquivos dos diretórios e subdiretórios da origem para o destino.
  - `-v` : Mostra os arquivos a medida que são copiados.
  - `-l` : Faz o *link* simbólico no destino ao invés de copiar os arquivos.

## mv

- Utilidade: Mover arquivos.
- Estrutura: `mv [opções] [origem] [destino]`
- Opções:
  - `-v` : Mostra os arquivos a medida que são movidos.

## find

- Utilidade: Buscar arquivos.
- Estrutura: `find [diretorio] [opções] [expressão]`

- Opções:
  - `-atime` : Procurar arquivos acessados dentro de um intervalo de dias.
  - `-mtime` : Procurar arquivos modificados dentro de um intervalo de dias.
  - `-ctime` : Procurar arquivos criados dentro de um intervalo de dias.
  - `-name` : Procura um arquivo pelo nome — é *case sensitive*.
  - `-iname` : Procura um arquivo pelo nome — não é *case sensitive*.
  - `-user` : Procura arquivos pelo usuário.
  - `-group` : Procura arquivos pelo grupo.
  - `-perm` : Procura arquivos pela permissão.
  - `-type` : Procura um arquivo pelo tipo. Os seguintes tipos são aceitos:
    - `f` : arquivo regular;
    - `d` : diretório;
    - `b` : bloco;
    - `c` : caractere;
    - `l` : *link* simbólico;
    - `p` : *pipe*;
    - `s` : *socket*.
- Macetes:
  1. Para fazer buscas com o `-atime`, `-mtime` ou `-ctime` é interessante não somente indicar os dias, mas também especificar o intervalo. O `-` significa "a menos de", enquanto o `+` significa "a mais de". Por exemplo, o comando `find / -atime -2` busca arquivos acessados a menos de dois dias.
  2. É possível concatenar as opções e fazer buscas mais refinadas. Por exemplo: `find / -iname teste -user root`.

## Referências globais

Coringas, ou referências globais, é um recurso usado para especificar um ou mais arquivos do sistema de uma só vez. Este recurso permite fazer a filtragem do que será listado, copiado, apagado, buscados e etc. No GNU/Linux são usados 4 tipos de coringas, são eles:

- `*` : Faz referência a um nome completo ou restante de um arquivo.
- `?` : Faz referência a uma letra naquela posição.
- `[ ]` : Faz referência a uma faixa de caracteres de um arquivo. Os padrões aceitos estão listados abaixo — caso qualquer um deles for precedido por um `^` significa que a expressão deve desconsiderar os caracteres indicados:
  - `[a-z][0-9]` : Faz referência aos caracteres de `a` até `z` seguido de um caractere de `0` até `9`.
  - `[a,z][1,0]` : Faz referência aos caracteres `a` e `z` seguido de um caractere `1` ou `0` naquela posição.
  - `[a-z,1,0]` : Faz referência ao intervalo de caracteres de `a` até `z` ou `1` ou `0` naquela posição.
- `{ }` : Expande e gera strings para pesquisa de padrões de um arquivo/diretório.
  - `x{ab,01}` : Faz referência a sequencia de caracteres `xab` ou `x01`.

- o `x{a-z,10}` : Faz referencia a sequencia de caracteres `xa-z` e `x10` .

## Compactadores

Compactadores são programas que diminuem o tamanho de um ou vários arquivo através de algoritmos muito avançados e complexos. A eficiência desses algoritmos pode ser mesurada pela taxa de compactação — que é o quanto um arquivo foi compactado. Por exemplo, se o tamanho do arquivo for diminuído a metade após a compactação, a taxa de compactação foi de 2:1 ou dois para um.

É importante frisar que não é possível trabalhar diretamente com arquivos compactados. Para manipula-los é necessário descompactar o arquivo. Entretanto, alguns programas descompactam o arquivo, o abrem e assim que o trabalho estiver concluído o compactam novamente, tudo isso para facilitar a rotina de trabalho.

## Tipos de compactação

Existem, basicamente, dois tipos de compactação, são elas:

- Compactação sem perda: Não causa nenhuma perda nas informações contidas no arquivo.
- Compactação com perda: É desenvolvida para atingir altas taxas de compactação, porém com perdas parciais dos dados. É aplicada a tipos de arquivos especiais, como músicas e imagens ou arquivos que envolvam a percepção humana.

## Arquivos de compactação

Para identificar um arquivo compactado e o programa utilizado para descompactá-lo, basta verificar a extensão do arquivo. Abaixo segue uma listagem dos programas mais comuns de descompactação e como utiliza-los:

- `gzip` : É capaz de compactar arquivos para a extensão `.gz` e descompactá-los. Possui uma ótima taxa de compactação e velocidade.
  - o Estrutura: `gzip [opções] [arquivos]` .
  - o Opções:
    - `-d` : Descompacta um arquivo.
    - `-l` Lista o conteúdo de um arquivo `.gz` .
    - `-r` : Compacta diretórios e subdiretórios.
- `bzip2` : É um novo compactador que vem sendo cada vez mais usado porque consegue atingir a melhor compactação em arquivos texto se comparado aos já existentes, em consequência sua velocidade de compactação também é menor; quase duas vezes mais lento que o `gzip` .
  - o Estrutura: `bzip2 [opções] [arquivos]` .
  - o Opções:
    - `-d` : Descompacta um arquivo.
    - `-l` Lista o conteúdo de um arquivo `.bz2` .



- `-r` : Compacta diretórios e subdiretórios.
- `zip` : Utilitário de compactação compatível com arquivos de extensão `.zip`.
  - o Estrutura: `zip [opções] [arquivo-destino] [arquivo-origem]` .
  - o Opções:
    - `-r` : Compacta diretórios e subdiretórios.
    - `-e` : Permite encriptar o conteúdo de um arquivo `.zip` através de senha. A senha será pedida no momento da compactação.
    - `-y` : Armazena links simbólicos no arquivo `.zip` . Por padrão, os links simbólicos são ignorados durante a compactação.
  - o Macetes:
    1. Para a descompactação de arquivos `.zip` no GNU/Linux, é necessário o uso do `unzip` seguido do nome do arquivo que deseja descompactar.
- `rar` : É um compactador que trabalha com arquivos de extensão `.rar` .
  - o Estrutura: `rar [ações] [opções] [arquivos]` .
  - o Ações:
    - `a` : Compacta arquivos.
    - `x` : Descompacta arquivos.
  - o Opções:
    - `p` : Inclui senha no arquivo. Cuidado, pessoas conectadas ao sistema operacional podem capturar a linha de comando facilmente e descobrir a senha.
- `tar` : Não é um compactador, ele apenas junta vários arquivos em um só.
  - o Estrutura: `tar [opções] [arquivo-destino] [arquivo-origem]` .
  - o Opções:
    - `-x` : Extrai arquivos `.tar` .
    - `-t` : Lista o conteúdo de um arquivo `.tar` .
    - `-c` : Cria um novo arquivo `.tar` .
  - o Macetes:
    1. É comum encontrar arquivos `.tar` compactados em um arquivo `.gz` — nesses casos a extensão do arquivo será `.tar.gz` . Para descompactá-lo basta utilizar a opção `-z` para chamar o `gzip` .

## Permissões de acesso

As permissões de acesso protegem o sistema de arquivos do GNU/Linux do uso indevido de pessoas ou programas não autorizados. Para proporcionar essa segurança, o GNU/Linux define o acesso aos arquivos por donos, grupos e outros usuários. A seguir, cada um deles serão explicados:

- **Dono:** É a pessoa que criou o arquivo, só ela pode modificar as permissões de acesso dele. O seu nome é o mesmo do usuário usado para entrar no GNU/Linux, mas também é possível identifica-lo por um número chamado *UID*, abreviação de *User ID* — para vê-lo basta executar o comando `id -u`.
- **Grupo:** Permite que vários usuários diferentes tenham acesso a um mesmo arquivo — cada usuário pode fazer parte de um ou mais grupos. Eles podem ser identificados pelo nome utilizado no momento de sua criação ou por um número chamado *GID*, abreviação de *Group ID* — para vê-lo basta executar o comando `id -g`.
- **Outros:** É a categoria de usuários que não são donos ou não pertencem ao grupo do arquivo.

## Tipos de permissões de acesso

No GNU/Linux existem três permissões básicas, são selas:

- **r** : Permissão de leitura para arquivos.
- **w** : Permissão de gravação para arquivos — também permite a exclusão do arquivo.
- **x** : Permite a execução do arquivo, caso seja um programa executável. Se o arquivo for um diretório, permite que seja acessado através do comando `cd`.

Essas permissões se aplicam ao dono, grupo e outros. Para visualizá-las basta utilizar a opção `-l` do comando `ls` — para mais detalhes desse comando, basta visitar a seção "Controle de Permissões" deste manual. Ao executá-lo, a saída obtida será algo semelhante a isto `-rw-r--r-- root root 13 May 19 16:01 teste`, onde:

```
graph TD
    Permissoes["-rw-r--r-"] --> DescricaoPermissoes["Tipo de arquivo e permissões"]
    NomeDono["root"] --> DescricaoNomeDono["Nome do dono do arquivo"]
    NomeGrupo["root"] --> DescricaoNomeGrupo["Nome do grupo do arquivo"]
    Data["13 May 19 16:01"] --> DescricaoData["Data de atualização"]
    NomeArquivo["teste"] --> DescricaoNomeArquivo["Nome do arquivo"]
```

Para poder ler as informações da primeira sequência de caracteres — Tipo de arquivo e permissões — é preciso saber que:

```
graph TD
    Permissoes["-rw-r--r-"] --> PermissoesArquivoCodigo["-"]
    PermissoesArquivoCodigo --> TipoDeArquivo["Tipo de arquivo"]
    Permissoes --> PermissoesDonoCodigo["rw-"]
    PermissoesDonoCodigo --> PermissoesDono["Permissões do dono"]
    Permissoes --> PermissoesGrupoCodigo["r-"]
    PermissoesGrupoCodigo --> PermissoesGrupo["Permissões do grupo"]
```

Permissoes --> PermissoesOutrosCodigo[r--] PermissoesOutrosCodigo--  
 >PermissoesOutros[Permissões de outros]  
 - O primeiro caractere representa o tipo de arquivo. Ele pode ser: - `-`: regular; -  
 `d`: diretório; - `l`: \*link\* simbólico; - `b`: bloco; - `c`: caractere; - `p`: \*pipe\*; - `s`:  
 \*socket\* - Do segundo ao quarto caractere, são as permissão de acesso do dono  
 do arquivo. - Do quinto ao sétimo caractere, são as permissão de acesso do  
 grupo do arquivo. - Do oitavo ao décimo caractere, são as permissão de acesso  
 dos outros usuários. ### Permissões na notação octal É possível escrever as  
 permissões com números por meio da notação octal. Ela é uma forma mais  
 objetiva de escrita, pois você especifica diretamente a permissão do dono, grupo  
 e outros. A correspondência dessa notação com a convencional é a seguinte: - 1  
 equivale ao `x`; - 2 equivale ao `w`; - 4 equivale ao `r`. Sabendo disso, já é  
 possível converter as permissões para a notação octal, veja o exemplo:  
 graph TD
 PermissoesConvencionais[-rwxrw-r--] --> PermissoesConvencionais
 PermissoesArquivoCodigo[-] --> PermissoesArquivoCodigo
 PermissoesArquivoCodigo --> TipoArquivo[Tipo de  
arquivo]
 PermissoesConvencionais --> PermissoesConvencionaisDono[rwx]
 PermissoesConvencionaisDono --> ConverterLeituraDono[r=4]
 PermissoesConvencionaisDono --> ConverteEscritaDono[w=2]
 PermissoesConvencionaisDono --> ConverterExecucaoDono[x=1]
 ConverterLeituraDono --> ConversaoPermissaoDono[4+2+1]
 ConverteEscritaDono --> ConversaoPermissaoDono
 ConverterExecucaoDono --> ConversaoPermissaoDono
 ConversaoPermissaoDono --> PermissoesConvencionais
 PermissoesConvencionaisGrupo[rw-] --> PermissoesConvencionaisGrupo
 ConverterLeituraGrupo[r=4] --> PermissoesConvencionaisGrupo
 ConverteEscritaGrupo[w=2] --> PermissoesConvencionaisGrupo
 ConverterExecucaoGrupo[x=0] --> ConverterLeituraGrupo
 ConverterLeituraGrupo --> ConversaoPermissaoGrupo[4+2+0]
 ConverteEscritaGrupo --> ConversaoPermissaoGrupo
 ConverterExecucaoGrupo --> ConversaoPermissaoGrupo
 ConversaoPermissaoGrupo --> PermissoesConvencionais
 PermissoesConvencionaisOutros[r--] --> PermissoesConvencionaisOutros
 ConverterLeituraOutros[r=4] --> PermissoesConvencionaisOutros
 ConverteEscritaOutros[w=0] --> PermissoesConvencionaisOutros
 ConverterExecucaoOutros[x=0] --> ConverterLeituraOutros
 ConverterLeituraOutros --> ConversaoPermissaoOutros[4+0+0]
 ConverteEscritaOutros --> ConversaoPermissaoOutros
 ConverterExecucaoOutros --> ConversaoPermissaoOutros
 ConversaoPermissaoOutros --> ConversaoPermissaoDono
 ConversaoPermissaoDono --> PermissoesOctal[764]
 ConversaoPermissaoGrupo --> PermissoesOctal
 ConversaoPermissaoOutros --> PermissoesOctal

- Primeiro: divide-se as permissões de usuário, grupo e outros;
- Segundo: Verificar quais permissões foram dadas para cada perfil e efetuar a conversão;
- Terceiro: Efetuar a soma das permissões dadas para cada perfil;
- Quarto: Agrupar o resultado do somatório de cada perfil para obter as permissões no formato octal.

## Manipulando permissões

A seguir serão listados os comandos utilizados para manipular as permissões dos arquivos.

## chmod

- Utilidade: Alterar as permissões de acesso.
- Estrutura: `chmod [opções] [permissões] [caminho/arquivo]` .
- Opções:
  - `-R` : Muda permissões de acesso do diretório atual e dos arquivos e subdiretórios contidos nele.
- Permissões: Para definir as permissões com a notação convencional é preciso informar três coisas:
  - O que deseja editar — dono( `u` ), grupo( `g` ) ou outro( `o` );
  - A ação que deseja efetuar — adicionar( `+` ), remover( `-` ) ou redefinir( `=` ) as permissões;
  - A permissão no formato convencional — leitura( `r` ), escrita( `w` ) ou execução( `x` ).
- Macetes:
  - É mais fácil editar permissões utilizando a notação octal, pois é possível alterar as permissões do dono, grupo ou outro. Para isso, basta colocar o código octal correspondente ao conjunto de permissões que deseja conceder.

## chown

- Utilidade: Muda dono de um arquivo — opcionalmente pode também ser usado para mudar o grupo.
- Estrutura: `chown [opções] [dono:grupo] [caminho/arquivo]` .
- Opções:
  - `-R` : Muda permissões de acesso do diretório atual e dos arquivos e subdiretórios contidos nele.

## setfacl

- Utilidade: permite personalizar permissões para usuários ou grupos específicos.
- Estrutura: `setfacl [opções] [mascara] [caminho/arquivo]` .
- Opções:
  - `-m` : indica o desejo de modificar usuário ou grupo que tem acesso ao arquivo;
  - `-x` : indica o desejo de excluir as permissões dadas para alguém.
  - `-R` : Muda permissões de acesso do diretório atual e dos arquivos e subdiretórios contidos nele.
- Mascara: Para definir as permissões é preciso informar:
  - O que será editado — usuário( `user:` ) ou grupo( `group:` );
  - Indicar o nome de quem receberá as permissões especiais, com `:` ao fim;



- Por fim, indicar as permissões dadas — podendo ser feito com a notação convencional quanto na octal.

## **getfacl**

- Utilidade: Visualizar usuários, grupos e suas respectivas permissões de um arquivo.
- Estrutura: `getfacl` `[opções]` `[mascara]` `[caminho/arquivo]` .

## Palavras-chaves

Esta seção apresenta um glossário com as palavras-chaves presentes neste manual. As explicações contidas aqui tem o objetivo de elucidar, rapidamente, a compreensão do leitor perante as informações trazidas.

<b>Termos e siglas</b>	<b>Definição e significado</b>
Sistema Operacional	É o conjunto de programas que fazem a interface do usuário e seus programas com o computador.
<i>Kernel</i>	É a base do sistema operacional. Ele faz a interface entre os programas e os recursos físicos do computador.
<i>Software</i>	Programas usados no computador.
GNU	Acrônimo recursivo para “GNU's Not Unix”. Trata-se do conjunto de ferramentas que compõem o sistema operacional GNU/Linux.
Linux	<i>Kernel</i> , baseado em Unix, utilizado pelo sistema operacional GNU/Linux.
<i>Open Source</i>	É o desenvolvimento baseado no compartilhamento e na melhoria colaborativa do código-fonte.
<i>Case Sensitive</i>	Capacidade de diferenciar letras minúsculas e maiúsculas.
<i>Script</i>	Texto contendo uma série de instruções sequenciais que podem ser executados por um interpretador.
Compilação	É, basicamente, a conversão de todas as linhas de um programa escrito em linguagem humana para uma linguagem de máquina.
<i>Buffer</i>	Área de armazenamento temporário de dados.
Processo	São tarefas em execução.
<i>Filesystem Hierarchy Standard (FHS)</i>	Padrão para sistema de arquivos hierárquico.
<i>Pendrives</i>	Dispositivos portáteis de armazenamento de arquivos.
<i>Compact Disc (CDs)</i>	Disco usado para armazenamento de arquivos.
<i>Digital Versatile Disc (DVDs)</i>	Evolução do CD para armazenamento de dados em discos.
<i>Blu-Ray</i>	Padrão de armazenamento de dados em disco superior ao DVD.
<i>Hardware</i>	Parte física do computador.
<i>Root</i>	Credencial concedida ao usuário que tem plena liberdade de acesso a qualquer arquivo ou diretório dos sistemas GNU/Linux. Esse termo, as vezes, é utilizado também para se referir ao diretório <code>/</code> do GNU/Linux.
<i>Web Servers</i>	Programas responsáveis por fazer a comunicação através de protocolos entre o computador que hospeda uma aplicação e o computador que acessa a aplicação.
<i>Drivers</i>	Programas que permitem que o sistema operacional e um dispositivo se comuniquem um com o outro.

<b>Termos e siglas</b>	<b>Definição e significado</b>
Imagem ISO	É uma cópia idêntica de arquivos gravados num <i>CD</i> , <i>DVD</i> ou <i>Blu-Ray</i> .
Comandos	Ordens passadas ao sistema operacional para executar uma determinada tarefa.
<i>Bytes</i>	Unidade de medida equivalente a <i>8bits</i> .
<i>Kbytes</i>	Unidade de medida equivalente a <i>1024Bytes</i> .
<i>Mbytes</i>	Unidade de medida equivalente a <i>1024Kbytes</i> .
<i>Gbytes</i>	Unidade de medida equivalente a <i>1024Mbytes</i> .
Nano	Programa de edição de texto.
Vim	Programa de edição de texto.



## Referências bibliográficas

Esta seção apresenta o conjunto das fontes efetivamente utilizadas na construção deste manual. Cada referência dessa bibliografia está organizada pelo tópico que a mesma ajudou a construir — para facilitar uma busca mais aprofundada do conteúdo.

## O que é GNU/Linux?

- Sistema Operacional. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch01s03.html>. Acesso em: 24 de jun. de 2021.
- History of the OSI. **Open Source Initiative**. Disponível em: <https://opensource.org/history/>. Acesso em: 24 de jun. de 2021.
- O que é GNU/Linux. **Viva o Linux**. Disponível em: <https://www.vivaolinux.com.br/linux/>. Acesso em: 24 de jun. de 2021.
- O que é Linux?. **Red Hat**. Disponível em: <https://www.redhat.com/pt-br/topics/linux/what-is-linux>. Acesso em: 24 de jun. de 2021.
- O que é GNU/Linux?. **Debian**. Disponível em: <https://www.debian.org/releases/stable/s390x/ch01s02.pt.html>. Acesso em: 24 de jun. de 2021.
- Visão Geral do Sistema GNU. **GNU**. Disponível em: <https://www.gnu.org/gnu/gnu-history.html>. Acesso em: 24 de jun. de 2021.

## As distribuições GNU/Linux

- Distribuições do Linux. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch01s05.html>. Acesso em: 24 de jun. de 2021.
- Hardware e Software. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch02.html#basico-hardsoft>. Acesso em: 06 de jul. de 2021.

## O interpretador de comandos

- Interpretador de comandos. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch02s08.html>. Acesso em: 06 de jul. de 2021.
- Comandos. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch02s05.html>. Acesso em: 06 de jul. de 2021.

## Terminal virtual

- Terminal Virtual (console). **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch02s09.html>. Acesso em: 06 de jul. de 2021.

## Uso do terminal

- Aprenda os comandos básicos do Linux - Terminal Ubuntu. **Diolinux**. Disponível em: <https://youtu.be/JEhVB4VHsTI>. Acesso em: 07 de jul. de 2021.

## Arquivos no GNU/Linux

- Arquivo. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch02s02.html>. Acesso em: 05 de jul. de 2021.

## Nomes e extensões

- Arquivo. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch02s02.html>. Acesso em: 05 de jul. de 2021.
- O que é um arquivo oculto?. **Fiodevida**. Disponível em:  
<https://fiodevida.com/o-que-e-um-arquivo-oculto/>. Acesso em: 05 de jul. de 2021.

## Tipos de arquivos

- Arquivo. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch02s02.html>. Acesso em: 05 de jul. de 2021.
- Tipos de Arquivos em Linux. **Portal GSTI**. Disponível em:  
<https://www.portalgsti.com.br/2017/04/tipos-de-arquivos-em-linux.html>. Acesso em: 13 de jul. de 2021.
- Métodos de tradução: interpretador x compilador. **iMasters**. Disponível em:  
<https://imasters.com.br/desenvolvimento/metodos-de-traducao-interpretador-x-compilador>. Acesso em: 05 de jul. de 2021.
- Existe diferença entre "script" e programa?. **Quora**. Disponível em:  
<https://pt.quora.com/Existe-diferen%C3%A7a-entre-script-e-programa>. Acesso em: 05 de jul. de 2021.
- O que são processos de um sistema operacional e por que é importante saber. **Tecmundo**. Disponível em:  
<https://www.tecmundo.com.br/memoria/3197-o-que-sao-processos-de-um-sistema-operacional-e-por-que-e-importante-saber.htm>. Acesso em: 13 de jul. de 2021.

## Diretórios no GNU/Linux

- Diretório. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch02s03.html>. Acesso em: 05 de jul. de 2021.
- Diretório. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch02s03.html>. Acesso em: 05 de jul. de 2021.

## Estrutura de diretórios do GNU/Linux

- Diretório. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch02s03.html>. Acesso em: 05 de jul. de 2021.
- Estrutura de diretórios do Linux: Compreenda de uma vez por todas. **Profissionais Linux**. Disponível em: <https://e-tinet.com/linux/estrutura-diretorios-linux/>. Acesso em: 05 de jul. de 2021.
- A Estrutura de Diretórios do Linux. **Bóson Treinamentos em Ciência e Tecnologia**. Disponível em: <http://www.bosontreinamentos.com.br/linux/a-estrutura-de-diretorios-do-linux/>. Acesso em: 05 de jul. de 2021.
- Conceito de CD **Conceito.de**. Disponível em: <https://conceito.de/cd>. Acesso em: 06 de jul. de 2021.
- Significado de DVD. **Significados**. Disponível em:  
<https://www.significados.com.br/dvd/>. Acesso em: 06 de jul. de 2021.
- O que é um Servidor Web (Web Server). **Melhores Hospedagem**. Disponível em: <https://www.melhoreshospedagem.com/servidor-web/>. Acesso em: 06 de jul. de 2021.
- O que é driver?. **Tecmundo**. Disponível em:  
<https://www.tecmundo.com.br/driver/207-o-que-e-driver-.htm>. Acesso em: 06 de jul. de 2021.
- O que é uma imagem ISO?. **Controle.net**. Disponível em:  
<https://www.controle.net/faq/o-que-e-uma-imagem-iso>. Acesso em: 06 de jul. de 2021.
- Hardware e Software. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch02.html#basico-hardsoft>. Acesso em: 06 de jul. de 2021.

## Manipulação de arquivos

- Capítulo 6. Comandos para manipulação de diretório. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch06.html>. Acesso em: 13 de jul. de 2021.
- Capítulo 7. Comandos para manipulação de Arquivos. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch07.html>. Acesso em: 13 de jul. de 2021.

### ls

- ls. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch06.html#comando-ls>. Acesso

em: 13 de jul. de 2021.

- Tamanho de arquivo. **GuiaFoca**. Disponível em <https://www.guiafoca.org/guiaonline/iniciante/ch02s02.html#basico-arquivo-tamanho>. Acesso em: 13 de jul. de 2021.

## cd

- cd. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch06s02.html>. Acesso em: 13 de jul. de 2021.

## pwd

- pwd. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch06s03.html>. Acesso em: 13 de jul. de 2021.

## mkdir

- mkdir. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch06s04.html>. Acesso em: 13 de jul. de 2021.

## rmdir

- rmdir. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch06s05.html>. Acesso em: 13 de jul. de 2021.

## cat

- cat. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch07.html#comandos-cat>. Acesso em: 13 de jul. de 2021.

## touch

- Aprenda a gerenciar arquivos no modo console do Linux. **CanalTech**. Disponível em: <https://canaltech.com.br/linux/Aprenda-a-gerenciar-arquivos-no-modo-console-do-Linux/>. Acesso em: 13 de jul. de 2021.

## rm

- rm. **GuiaFoca**. Disponível em: <https://www.guiafoca.org/guiaonline/iniciante/ch07s03.html>. Acesso em: 13 de jul. de 2021.

## cp

- cp. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch07s04.html>. Acesso em: 13 de jul. de 2021.

## mv

- mv. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch07s05.html>. Acesso em: 13 de jul. de 2021.

## find

- find. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch08s06.html>. Acesso em: 13 de jul. de 2021.
- Comandos para procurar arquivos no Linux - FIND e LOCATE. **LINUXtips**. Disponível em: <https://www.youtube.com/watch?v=ARpruwUhKPA>. Acesso em: 13 de jul. de 2021.

## Referências globais

- Coringas. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/iniciante/ch02s12.html>. Acesso em: 13 de jul. de 2021.

## Compactadores

- Capítulo 18. Compactadores. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/intermediario/ch18.html>. Acesso em: 14 de jul. de 2021.

### Tipos de compactação

- Tipos de compactação. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/intermediario/ch18.html#cpctd-tipos>. Acesso em: 14 de jul. de 2021.

### Arquivos de compactação

- Extensões de arquivos compactados. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/intermediario/ch18s02.html>. Acesso em: 14 de jul. de 2021.
- gzip. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/intermediario/ch18s03.html>. Acesso em: 14 de jul. de 2021.
- bzip. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/intermediario/ch18s07.html>. Acesso em: 14 de jul. de 2021.

- zip. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/intermediario/ch18s04.html>. Acesso em:  
14 de jul. de 2021.
- rar. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/intermediario/ch18s08.html>. Acesso em:  
14 de jul. de 2021.
- tar. **GuiaFoca**. Disponível em:  
<https://www.guiafoca.org/guiaonline/intermediario/ch18s06.html>. Acesso em:  
14 de jul. de 2021.