



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO PAULO

Yhasmin Souza e Silva

SP3217027

Análise e Desenvolvimento de Sistemas

SPOSOPE

## Comandos de Manipulação de Arquivos e Diretórios, Permissões e Propriedades

### 1. Introdução

A manipulação de arquivos e diretórios é uma das tarefas mais fundamentais em qualquer sistema operacional. Desde ações básicas como criação e exclusão de arquivos até operações mais avançadas, como alteração de permissões e análise de propriedades, o domínio desses comandos é essencial tanto para usuários quanto para administradores de sistemas. Este trabalho apresenta uma visão geral dos principais comandos utilizados para manipulação de arquivos e diretórios, bem como das permissões e propriedades que regulam o acesso a esses elementos.

### 2. Comandos de Manipulação de Arquivos e Diretórios

#### 2.1 Comando pwd

O comando pwd (print working directory) exibe o diretório de trabalho atual do usuário. É utilizado para localizar em qual pasta o usuário se encontra no sistema.

##### Exemplo:

```
pwd
```

Saída:

```
/home/usuario
```

#### 2.2 Comando ls

Utilizado para listar o conteúdo de diretórios. Pode ser combinado com parâmetros para exibição detalhada.

##### Exemplos:

```
ls          # Lista os arquivos e pastas
ls -l       # Exibe em formato detalhado
ls -a       # Lista arquivos ocultos
```

#### 2.3 Comando cd

O comando cd (change directory) altera o diretório atual do usuário.

##### Exemplos:

```
cd /home/usuario/documentos
```

```
cd ..      # Volta um nível no diretório
```

## 2.4 Comando mkdir

Cria um ou mais diretórios.

### Exemplos:

```
mkdir projetos  
mkdir -p projetos/python/exercicios
```

## 2.5 Comando rmdir

Remove diretórios vazios.

### Exemplo:

```
rmdir pasta_vazia
```

## 2.6 Comando rm

Remove arquivos e diretórios. Com o parâmetro -r, é possível remover diretórios e seus conteúdos recursivamente.

### Exemplos:

```
rm arquivo.txt  
rm -r pasta
```

## 2.7 Comando cp

Copia arquivos e diretórios. Com o parâmetro -r, diretórios são copiados de forma recursiva.

### Exemplos:

```
cp arquivo.txt copia.txt  
cp -r pasta_original nova_pasta
```

## 2.8 Comando mv

Move ou renomeia arquivos e diretórios.

### Exemplos:

```
mv antigo.txt novo.txt  
mv pasta1/arquivo.txt pasta2/
```

## 2.9 Comando touch

Cria arquivos vazios ou atualiza a data de modificação de arquivos existentes.

### Exemplo:

```
touch novo_arquivo.txt
```

## 2.10 Comando cat

Exibe o conteúdo de arquivos de texto diretamente no terminal.

**Exemplo:**

```
cat arquivo.txt
```

## 3. Permissões e Propriedades de Arquivos e Diretórios

No sistema de arquivos, cada arquivo e diretório possui permissões associadas que controlam o acesso por parte de usuários e grupos. Tais permissões estão divididas em três categorias:

- **Usuário (owner):** O criador ou proprietário do arquivo.
- **Grupo (group):** Um grupo de usuários associados ao arquivo.
- **Outros (others):** Todos os demais usuários.

### 3.1 Tipos de Permissões

As permissões são:

- **r (read):** Permite leitura do conteúdo do arquivo ou listagem de diretórios.
- **w (write):** Permite modificar o conteúdo do arquivo ou adicionar/remover arquivos no diretório.
- **x (execute):** Permite a execução de arquivos ou o acesso ao diretório.

### 3.2 Visualização de Permissões

Com o comando `ls -l`, as permissões são apresentadas da seguinte forma:

```
-rwxr-xr--
```

Nessa notação:

- `rwx` (usuário): leitura, escrita e execução.
- `r-x` (grupo): leitura e execução.
- `r--` (outros): somente leitura.

### 3.3 Modificação de Permissões (chmod)

O comando `chmod` (change mode) altera as permissões de arquivos e diretórios. Pode ser usado de forma simbólica ou numérica.

**Exemplos:**

```
chmod 755 script.sh # rwxr-xr-x
```

```
chmod u+x programa # adiciona permissão de execução ao usuário
```

### 3.4 Modificação de Propriedades (chown e chgrp)

- **chown:** Altera o proprietário de um arquivo.

- **chgrp**: Altera o grupo associado ao arquivo.

**Exemplo:**

```
chown maria arquivo.txt  
chown maria:desenvolvedores script.sh
```

#### 4. Considerações Finais

Compreender e saber utilizar corretamente os comandos de manipulação e as permissões de arquivos e diretórios é fundamental para a segurança e organização de sistemas computacionais. Esses comandos constituem a base para tarefas mais complexas no gerenciamento de sistemas operacionais, especialmente em ambientes Unix e Linux.

#### 5. Referências

DEVMedia. *Comandos básicos do Linux: manipulando diretórios.*

<https://www.devmedia.com.br/comandos-basicos-do-linux-manipulando-diretorios/18118>. Acesso em: 6 abr. 2025.

WIKILIVROS. *Linux Essencial/Lição Manipulação de arquivos.*

[https://pt.wikibooks.org/wiki/Linux\\_Essencial/Li%C3%A7%C3%A3o\\_Manipula%C3%A7%C3%A3o\\_de\\_arquivos](https://pt.wikibooks.org/wiki/Linux_Essencial/Li%C3%A7%C3%A3o_Manipula%C3%A7%C3%A3o_de_arquivos). Acesso em: 6 abr. 2025.

ESCOLA DNC. *Guia completo de comandos Linux: navegação, manipulação de arquivos e permissões.*

<https://www.escoladnc.com.br/blog/guia-completo-de-comandos-linux-navegacao-manipulacao-de-arquivos-e-permissoes/>. Acesso em: 6 abr. 2025.

CERTIFICAÇÃO LINUX. *Permissões e propriedades de arquivos.*

<https://www.certificacaolinux.com.br/permissoes-e-propriedades-de-arquivos/>. Acesso em: 6 abr. 2025.

MTM UFSC. *Comandos para manipulação de arquivos - Guia Foca GNU/Linux.*

<https://mtm.ufsc.br/~krukoski/pub/linux/focalinux1/ch-cmd.htm>. Acesso em: 6 abr. 2025.

MEDIUM. *Terminal no Linux: manipulação de arquivos.*

<https://medium.com/21-blog/terminal-no-linux-manipula%C3%A7%C3%A3o-de-arquivos-52519c7444f9>. Acesso em: 6 abr. 2025.

YOUTUBE. *Permissões em arquivos e diretórios - Linux Ubuntu e derivados.*

<https://www.youtube.com/watch?v=aFV7oBEWBPM>. Acesso em: 6 abr. 2025.