

Administração de Sistemas Operacionais Linux

O Sistema de Arquivos

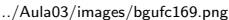
Antonio Rafael Braga

Universidade Federal do Ceara

11 de julho de 2025

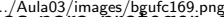
../Aula03/images/slide_capa_169.jpg

Agenda

- 1 Introdução
- 2 Pathnames
- 3 Organização da Árvore de Arquivos
- 4 Tipos de Arquivos
../Aula03/images/bgufc169.png
- 5 Atributos de Arquivos
- 6 Montagem e desmontagem de Sistema de Arquivos
- 7 Atividades laboratório

Introdução

O sistema de arquivos pode ser considerado por meio de quatro componentes principais:


- Um *namespace* - uma maneira de nomear as coisas e organizá-las em uma hierarquia;
- Uma API - um conjunto de chamadas de sistema para navegar e manipular objetos;
- Modelos de segurança -  esquemas para proteger, ocultar e compartilhar coisas;
- Uma implementação - software para “amarrar” o modelo lógico ao hardware.

Pathnames

O sistema de arquivos é apresentado como uma única hierarquia unificada que começa no diretório raiz / e continua descendo através de um número arbitrário de subdiretórios.

- A lista de diretórios que devem ser percorridos para localizar um arquivo específico mais o nome do arquivo forma um nome de caminho.
../Aula03/Images/figure1001.png
- Os nomes de caminho podem ser absolutos (por exemplo, **/tmp/foo**) ou relativos (por exemplo, **book4/filesystem**).
- Os nomes de caminho relativos são interpretados a partir do diretório atual.

Pathnames

- O nome do arquivo e o caminho podem ser usados tanto para caminhos absolutos quanto relativos;
- Cada componente de um nome de caminho (ou seja, cada diretório) deve ter um nome com no máximo 255 caracteres;

- Limite no comprimento total do caminho que você pode passar para o kernel como um argumento de chamada do sistema (4.095 bytes).

Organização da Árvore de Arquivos

Fonte: [Evi Nemeth, 2017]

../Aula03/images/bgufc169.png

Organização da Árvore de Arquivos

Fonte: [Evi Nemeth, 2017]

../Aula03/images/bgufc169.png

Tipos de Arquivos

../Aula03/images/bgufc169.png

Tipos de Arquivos

comando file

Você pode determinar o tipo de um arquivo existente com o comando file.

```
$ file /usr/include
```

```
/usr/include/: directory
```

```
$ file /bin/sh
```

```
/bin/sh: symbolic link to dash
```

Arquivos regulares

Definição:

Os arquivos regulares consistem em uma série de bytes; os sistemas de arquivos não impõem nenhuma estrutura em seus conteúdos. Arquivos de texto, arquivos de dados, programas executáveis e bibliotecas compartilhadas são armazenados como arquivos regulares. O acesso sequencial e o acesso aleatório são permitidos.

Diretórios

Definição:

Um diretório contém referências nomeadas para outros arquivos.

Você pode criar diretórios com **mkdir** e excluí-los com **rmdir**.

As entradas especiais “.” e “..” referem-se ao próprio diretório e ao seu diretório pai;

Links físicos

Você cria links físicos com **ln** e os remove com **rm**. É fácil lembrar a sintaxe de **ln** se você tiver em mente que ela reflete a sintaxe de **cp**. O comando **cp oldfile newfile** cria uma cópia de **oldfile** chamada **newfile**, e **ln oldfile newfile** torna o nome **newfile** uma referência adicional ao **oldfile**.

Arquivos de dispositivos de caracteres e blocos

Importante:



Os drivers de dispositivo apresentam uma interface de comunicação padrão que se parece com um arquivo normal. Quando o sistema de arquivos recebe uma solicitação que se refere a um caractere ou arquivo de dispositivo de bloco, ele simplesmente passa a solicitação para o driver de dispositivo apropriado [Evi Nemeth, 2017].

Os arquivos são apenas pontos de encontro que se comunicam com os *drivers*!

Links Simbólicos

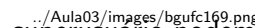
Importante!

A diferença entre links físicos e links simbólicos é que um link físico é uma referência direta, enquanto um link simbólico é uma referência por nome. Links simbólicos são distintos dos arquivos para os quais eles apontam.

Exemplo

```
$ sudo ln -s archived/secure /var/data/secure
```

Atributos de Arquivos

- O sistema tradicional de controle de acesso de proprietário/grupo/outros de 9 bits é poderoso o suficiente para acomodar a grande maioria das necessidades administrativas.
- O proprietário do arquivo e o  superusuário podem modificar os doze bits com o comando **chmod**.

Codificação de permissão para chmod

Octal	Binary	Perms	Octal	Binary	Perms
0	000	rWX	4	100	-WX
1	001	rW-	5	101	-W-
2	010	r-X	6	110	--X
3	011	r--	7	111	---

Exemplos de sintaxe mnemônica de chmod

Spec	Meaning
u+w	Adds write permission for the owner of the file
ug=rw,o=r	Gives r/w permission to owner and group, and read permission to others
a-x	Removes execute permission for all categories (owner/group/other)
ug=srx,o=	Makes setuid/setgid and gives r/x permission to only owner and group
g=u	Makes the group permissions be the same as the owner permissions

chown e chgrp: alterar proprietário e grupo

Funções

O comando **chown** altera o proprietário de um arquivo e o comando **chgrp** altera o proprietário do grupo.

O **chown** pode alterar o proprietário e o grupo de um arquivo de uma vez com a sintaxe **chown** user:group file ...

Por exemplo,

```
$ sudo chown -R matt:staff matt/restore
```

Montagem e desmontagem de Sistema de Arquivos

Função:

Os sistemas de arquivos são anexados à árvore com o comando **mount** que mapeia um diretório dentro da árvore de arquivos existente, chamado de ponto de montagem, para a raiz do recém-anexado sistema de arquivo.

Por exemplo,

```
$ sudo mount /dev/sda4 /users
```

O arquivo **/etc/fstab** lista os sistemas de arquivos que normalmente são montados no sistema.

../Aula03/images/bgufc169.png

Atividades laboratório

- 1 Crie um diretório chamado AulaLinux, em seguida entre no diretório e crie o arquivo chamado vozao1adivisao.txt. Obtenha informações dos elementos criados anteriormente.
- 2 Crie um link físico para o arquivo que você criou anteriormente. Copie o arquivo que você criou para outro diretório e altere o seu conteúdo. Verifique se o link físico que você criou foi alterado.

../Aula03/images/bgufc169.png
- 3 Crie um link simbólico para um arquivo que esteja em outro diretório.
- 4 Crie um link simbolico e delete o arquivo ao qual ele apontava. O que aconteceu ao tentar acessar o link simbólico? Crie o arquivo deletado novamente e verifique o que ocorreu ao tentar acessar o link simbólico novamente.
- 5 Faça o mesmo procedimento para um exemplo com link físico

Atividades laboratório 2

- 1 Crie um diretório chamado AulaArquivos, e o subdiretorio testechown. Altere o usuário e o grupo para algum que estejam no seu Linux.

- 2 `../Aula03/images/bgufc169.png`

Referências

[Evi Nemeth, 2017] Evi Nemeth, Garth Snyder, T. H. B. W. D. M. (2017).
Unix and Linux System Administration Handbook.
5th edition.

../Aula03/images/bgufc169.png