

Introdução ao Linux

Módulo 4

PERMISSÕES





Vamos aprender sobre permissões

Informações de permissão obtidas a partir do comando `ls -l`

```
drwxr-xr-x 3 ead ead 4096 jan 29 10:49 Documentos  
-rw-r--r-- 3 ead ead 4096 jan 29 10:49 arquivo_texto
```

A primeira coluna (**d** e **-**) correspondem ao tipo de arquivo:

- → arquivo comum
- d** → diretório
- c** → dispositivo de caractere
- b** → dispositivo de bloco
- s** → socket local
- p** → pipe nomeado (named pipe)
- l** → link simbólico

Os três grupos seguintes, compostos de três colunas cada, identificam as permissões para usuário, grupo e outros, respectivamente.

No exemplo acima, as permissões são:

O diretório **Documentos** tem as permissões **rw****x**, para o usuário proprietário do arquivo, **r-x**, para o grupo e **r-x** para todos outros usuários e grupos;

O arquivo **arquivo_texto** tem as permissões **rw**, para o usuário proprietário do arquivo, **r--**, para o grupo e **r--** para todos outros usuários e grupos.

As permissões **r**, **w** e **x** significam:

- → Sem permissão alguma;
- r** → Permissão de leitura. É possível ler o conteúdo de um arquivo ou ver o conteúdo de um diretório;
- w** → Permissão de escrita. É possível editar um arquivo ou criar arquivos dentro de um diretório;
- x** → Permissão de execução. É possível executar o arquivo caso seja um binário ou script, e acessar um diretório. Diretórios precisam da permissão de execução para que possam ser acessados independentemente de terem permissão de leitura.



Outras informações do ls -l

```
drwxr-xr-x 3 ead ead 4096 jan 29 10:49 Documentos
```

3 → é o número de diretórios dentro de Documentos

ead → (primeiro ead) o dono do arquivo/diretório

ead → (segundo ead) o grupo ao qual pertence

4096 → tamanho em bytes

jan 29 10:49 → data e hora da última modificação

Documentos → Nome do diretório ou arquivo, se for o caso

Mudando as permissões

```
<opções> <permissões> <arquivo/diretório>
```

São dadas por um número de 3 algarismos ABC, onde **A** é o somatório das permissões para o dono do arquivo, **B** para seu grupo e **C** para outros usuários

Permissões no modo **octal**:

0 sem permissão nesta posição

4 permissão de leitura

2 permissão de escrita

1 permissão de execução

Assim, se quisermos dar permissão de leitura (**4**) e escrita (**2**) para o dono do arquivo, e somente leitura (**4**) para o grupo e para os outros usuários, teremos:

A (dono do arquivo) → $4 + 2 = 6$

B (grupo) → 4

C (outros) → 4

Permissão em octal: ABC → 644

Exemplos:

chmod 700 arquivo → Dá todas permissões ao dono do arquivo, isto é, leitura (4), escrita (2) e execução (1), e nenhuma (0) ao grupo e outros usuários;

chmod 644 arquivo → Dá permissão de leitura (4) e escrita (2) para o dono do arquivo, e leitura (4) para o grupo e outros usuários;

chmod 755 diretório → Dá permissão de leitura (4), escrita (2) e execução (1) para o dono do diretório, e leitura (4) e execução (1) para o grupo e outros usuários.



Mudando propriedade de arquivos e diretórios

```
chown <opções> <dono><grupo> <arquivo ou diretório>
```

Exemplos:

chown ead meu_arquivo → dá ao usuário “ead” a propriedade do arquivo “meu_arquivo”;

chown ead:users meu_arquivo → dá ao usuário “ead” e ao grupo “users” a propriedade do arquivo “meu_arquivo”;

chown -R ead:users meu_diretório → dá ao usuário “ead” e ao grupo “users” a propriedade do diretório “meu_diretório” e de todo seu conteúdo (opção -R).

Informações detalhadas

Consulte as páginas de manual dos comandos digitando no terminal os comandos abaixo:

man chmod → manual do comando chmod

man chown → manual do comando chown

Créditos

Conteúdo

Glaucio Souza

Diagramação

Beatriz Maciel

Gustavo Maluf

Coordenação

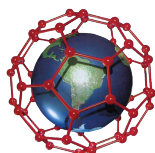
Daisy Schneider

Revisão

Diane Couto

Apoio

Editais UFRGS EAD 25



CESUP
Centro Nacional de Supercomputação



**Rede
Multivídeos**