

## Comandos Linux

### File System, Processos, Threads, System Info, Hardware

#### Comandos mais usuais (1)

**ls**: Lista todos os arquivos do diretório (list)

**ls -l**: Lista o tipo de arquivo e permissões

**df**: Mostra quantidade de espaço usada no disco rígido (disk file)

**top**: Mostra os processos consumindo memória

**cd**: Acessa uma determinada pasta (diretório) como por exemplo cd diretório, cd .., cd /

**mkdir**: Cria um diretório

**rm**: Remove um arquivo/diretório

**cat**: Abre um arquivo

exemplo: **cat > texto1.txt** (dê enter)

comandos para certificação lpi

(dê um CTRL +D)

**cat texto1.txt**

aparece o texto que foi salvo no arquivo txt

**vi**: Abre o editor vi para **editar/criar** arquivos

#### Comandos de Controle e Acesso (2)

**exit**: Terminar a sessão, ou seja, bash

**logout**: Deslogar, ou seja, terminar a sessão atual, mas apenas na bash shell

**passwd**: Mudar a password do nosso utilizador (usuário logado)

**ssh**: Sessão segura, vem de secure shell, e permite-nos logar num servidor remoto através do protocolo ssh

### Comandos de Ajuda e Documentação (3)

**apropos:** Localiza comandos por pesquisa de palavra-chave

**info:** Abre o explorador de informações

**man:** Manual muito completo, pesquisa informação acerca de todos os comandos que necessitemos de saber, como por exemplo man find

**whatis (o que é):** Descreve o que um determinado comando é/faz

**whereis (onde é):** Localizar a página de ajuda (man page), código fonte, ou arquivos binários, de um determinado programa.

### Comandos de Edição de Texto (4)

**emacs:** Editor de texto screen-oriented

**nano:** Editor de texto screen-oriented

**sed:** Editor de texto stream-oriented

**vi:** Editor de texto full-screen

**vim:** Editor de texto full-screen melhorado (vi improved)

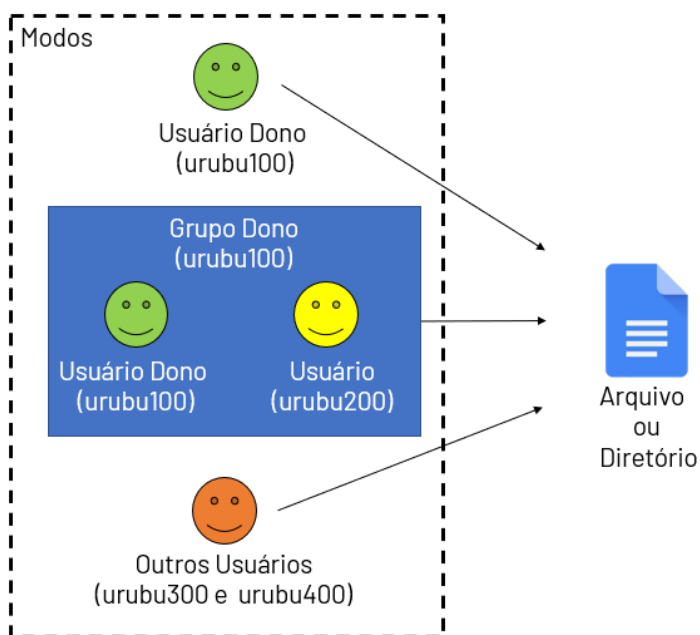
## Comandos de Gestão de Arquivos e Diretórios (5)

**chmod:** Mudar a proteção de um arquivo ou diretório, como por exemplo `chmod 777` (permissão total)

```
urubu100@DESKTOP-OQ1491A:~$ ls -l
total 0
drwxr-xr-x 1 root    root    512 Sep 18 01:13 apipy1
-rw-r--r-- 1 urubu100 urubu100 15 Sep 21 16:49 arqu1.txt
-rw-r--r-- 1 urubu100 urubu100 19 Sep 21 17:10 texto1.txt
urubu100@DESKTOP-OQ1491A:~$ chmod ug+rw texto1.txt
urubu100@DESKTOP-OQ1491A:~$ ls -l
total 0
drwxr-xr-x 1 root    root    512 Sep 18 01:13 apipy1
-rw-r--r-- 1 urubu100 urubu100 15 Sep 21 16:49 arqu1.txt
-rw-rw-r-- 1 urubu100 urubu100 19 Sep 21 17:10 texto1.txt
```

- **r** = leitura.
- **w** = gravação.
- **x** = execução (para arquivos) ou autorização de acesso (para diretórios).
- **u** = as permissões do dono do arquivo. (**u**ser owner)\*
- **g** = as permissões do grupo. (**g**roup)\*
- **o** = as permissões dos outros usuários do sistema. (**o**thers)\*
- **a** = todos os usuários do sistema. (**a**ll)\*

\*A combinação das letras ugoa no comando chmod define quais os usuários estão tendo as suas permissões de acesso alteradas.



drwxrwxrwx

d = Directory  
r = Read  
w = Write  
x = Execute

chmod 777

rwx | rwx | rwx  
Owner | Group | Others

Octal		Binário
7	rwX	111
6	rw-	110
5	r-X	101
4	r--	100
3	-wX	011
2	-w-	010
1	--X	001
0	---	000

**cmp:** Compara dois arquivos

**comm:** Seleciona ou rejeita linhas comuns a dois arquivos selecionados

**cp:** Copia arquivos, como o copy do MS-DOS

**crypt:** Encripta ou Descripta arquivos (apenas CCWF)

**diff:** Compara o conteúdo de dois arquivos ASCII

**file:** Determina o tipo de arquivo (exemplo: file "arquivo existente")

**grep:** Procura um arquivo por um padrão, sendo um filtro muito útil e usado, por exemplo um `cat a.txt | grep ola` irá mostrar-nos apenas as linhas do arquivo a.txt que contenham a palavra "ola" – ex: `grep "texto" arquivo.txt`

O comando **grep** imprime na tela as linhas que correspondem a um padrão em cada arquivo.

Um bom argumento para se utilizar (**man grep**) é o `-i` que ignora a distinção de letras maiúsculas e minúsculas.

```
eduardo@DESKTOP-LENOVO: /home/so_2adsa/20210323$  
eduardo@DESKTOP-LENOVO:/home/so_2adsa/20210323$ grep -i "EOF" *.txt  
file4.txt:Testando o cat e o EOF via root EOF  
instructionsCat.txt:para cat > arquivo << EOF "texto" EOF  
instructionsCat.txt:Obs.: EOF deriva de End-of-File que mostra um end of input  
eduardo@DESKTOP-LENOVO:/home/so_2adsa/20210323$
```

**gzip:** Comprime ou expande arquivo – ex: `gzip arquivo`

**ls:** Lista o conteúdo de uma diretório, semelhante ao comando **dir** no MS-DOS

**lsuf:** Lista os arquivos abertos, vem de list open files

**mkdir:** Cria uma diretório, vem de make directory

**mv:** Move ou renomeia arquivos ou diretórios

**rm:** Apaga arquivos, vem de remove, e é semelhante ao comando `del` no MS-DOS, é preciso ter cuidado com o comando `rm *` pois apaga tudo sem confirmação por defeito

**rmdir:** Apaga diretório, vem de remove directory

**stat:** Mostra o estado de um arquivo, útil para saber por exemplo a hora e data do último acesso ao mesmo

**sort:** Ordena, une ou compara texto, podendo ser usado para extrair informações dos arquivos de texto ou mesmo para ordenar dados de outros comandos como por exemplo listar arquivos ordenados pelo nome

**wc:** Conta linhas, palavras e mesmo caracteres num arquivo

**xv:** Serve para exibir, imprimir ou mesmo manipular imagens

**gv:** Exibe arquivos ps e pdf

**xpdf:** Exibe arquivos pdf, usa o gv

### Comandos de Transferência de Arquivos (6)

**ftp:** Vem de file transfer protocol, e permite-nos, usando o protocolo de transferência de arquivos ftp, transferir arquivos entre vários hosts de uma rede, como a um servidor de ftp para enviar ou puxar arquivos

### Comandos de Notícias ou Rede (7)

**netstat:** Mostra o estado da rede – netstat --route

**ssh:** protocolo de comunicação

**ifconfig:** Visualizar os ips da nossa máquina, entre outras funções relacionadas com ips

**ping:** Pingar um determinado host, ou seja, enviar pacotes icmp para um determinado host e medir tempos de resposta, entre outras coisas