



## Gerenciamento de Processos

Instituto Federal de Educação, Ciência e Tecnologia do Triângulo Mineiro

Prof. Edwar Saliba Júnior

Outubro de 2025



## O que são processos?

- De acordo com Tanenbaum *apud* Romero (2014), *"a ideia-chave aqui é que um processo é um tipo de atividade. Ele tem um programa, entrada, saída e um estado. Um único processador pode ser compartilhado entre vários processos, com algum algoritmo de agendamento sendo utilizado para determinar quando parar de trabalhar em um processo e servir a um diferente."*



## O que são processos?

- Um processo pode ser visto como a representação de um programa em execução, utilizando os recursos do computador para realizar alguma tarefa.
- Cada processo no GNU/Linux recebe um número para sua identificação, conhecido como **PID** (*Process Identifier*); esse número pode ser visto executando-se o comando **ps aux** ou **top**.



## O Processo “init”

- Quando inicializamos o Linux, o processo **init** é o primeiro a ser criado e é conhecido como o pai de todos os outros processos;
- para cada programa aberto, é criado um processo no sistema operacional e
- o processo **init** é o responsável por inicializar todos eles e o seu **PID** é o **1**.



## Verificando Processos

- Para verificarmos os processos de forma dinâmica, basta executarmos o comando:  
`top` ou `htop` (tecle **q** para encerrar e **F1** para ajuda)
- agora, para verificarmos listas de processos, podemos utilizar o comando `ps` com seus diversos parâmetros:  
`ps`  
`ps aux`  
`ps -A | grep apache` (`grep` filtra resultados)  
`ps axms` e etc.
- para contar quantos processos estão sendo executados no sistema operacional, utilize o comando: `ps aux | wc -l`
- para mais informações `man ps`



## Sinais de Processo

- Os sinais são uma forma de comunicação usada pelos processos para que o sistema consiga interferir em seu funcionamento.
- Ao receber um sinal com instruções, um processo interpreta e executa a ação que foi especificada no sinal.



## Sinais de Processo

- Alguns dos sinais mais conhecidos e usados por processos são:
  - **KILL** – sinal com função de encerrar um processo;
  - **TERM** – termina um processo após ele finalizar uma tarefa;
  - **STOP** – interrompe um processo em execução e
  - **CONT** – reativa a execução de um processo que tenha sido interrompido.



## Parando e Retomando Processos

- Para parar um processo, podemos utilizar o comando **kill** da seguinte maneira:  
**sudo kill -STOP PID**
- mas, se um determinado software tem diversos processos sendo executados simultaneamente, podemos fazer da seguinte maneira:  
**sudo killall -STOP nomeDoProcesso**
- e, de forma inversa, podemos retomar a execução de um processo interrompido, respectivamente, da seguintes maneiras:  
**sudo kill -CONT PID** ou  
**sudo killall -CONT nomeDoProcesso**





## Parando e Retomando um Processo

- Exemplo:
  - abrir o GEdit e
    - 1) digitar qualquer coisa
    - 2) pular para Terminal – passo 1
    - 3) tentar digitar qualquer coisa
    - 4) voltar ao Terminal – passo 4
    - 5) digitar qualquer coisa
  - abrir o Terminal lado a lado ao GEdit:
    - 1) `ps -A | grep gedit`
    - 2) `sudo kill -STOP PID_do_GEdit`
    - 3) voltar ao GEdit – passo 3
    - 4) `sudo kill -CONT PID_do_GEdit`
    - 5) voltar ao GEdit – passo 5



## Parando e Retomando Processos

- Exemplo (2 terminais lado a lado):
  - em um terminal:
    - 1) `sudo -u postgres psql`
    - 2) `\l` (listar bancos)
    - 3) `\c nomeBanco` (acessar banco)
    - 4) `\dt` (listar tabelas)
    - 5) `select...`
    - 6) pular para o outro terminal – passo 1
    - 7) `select...`
    - 8) pular para o outro terminal – passo 4
  - em outro terminal:
    - 1) `ps -A | grep postgres`
    - 2) `sudo killall -STOP postgres`
    - 3) voltar ao outro terminal – passo 7
    - 4) `sudo killall -CONT postgres`
    - 5) observar o outro Terminal.



## Matando um Processo

- Exemplo:
  - abrir o GEdit e
  - em um terminal:

```
ps -A | grep gedit  
sudo kill -9 PID_do_GEdit
```



## Prioridades dos Processos

- No Linux, cada processo tem uma prioridade de agendamento que ajuda o sistema a decidir quem usa mais tempo de CPU.
- Essa prioridade é representada por um valor "nice" de **-20** até **+19**, onde:
  - **-20** significa "prioridade máxima" (processo mais "egoísta")
  - **0** significa "prioridade padrão" e
  - **+19** significa "prioridade mínima" (processo mais "educado").



# Sistema Operacional GNU/Linux

## top

```
Terminal
top - 21:24:43 up 2:33, 1 user, load average: 1.30, 1.15, 1.14
Tasks: 319 total, 1 running, 318 sleeping, 0 stopped, 0 zombie
%Cpu(s): 16.1 us, 3.9 sy, 2.5 ni, 77.2 id, 0.3 wa, 0.0 hi, 0.1 si, 0.0 st
MiB Mem : 15913.4 total, 619.0 free, 10872.7 used, 5663.0 buff/cache
MiB Swap: 16307.0 total, 15887.0 free, 420.0 used, 5040.7 avail Mem
```

| PID   | USER  | PR | NI  | VIRT    | RES    | SHR    | S | %CPU | %MEM | TIME+    | COMMAND         |
|-------|-------|----|-----|---------|--------|--------|---|------|------|----------|-----------------|
| 5180  | edwar | 20 | 0   | 3127164 | 330036 | 86152  | S | 18.2 | 2.0  | 27:12.53 | WebExtensions   |
| 3161  | edwar | 20 | 0   | 5095996 | 359008 | 105912 | S | 12.3 | 2.2  | 5:23.22  | gnome-shell     |
| 7180  | edwar | 39 | 19  | 1358048 | 761636 | 17608  | S | 10.3 | 4.7  | 35:41.69 | tracker-miner-f |
| 4534  | edwar | 20 | 0   | 12.2g   | 748836 | 197768 | S | 9.9  | 4.6  | 24:42.79 | firefox-esr     |
| 2929  | edwar | 20 | 0   | 1333584 | 165236 | 92728  | S | 8.6  | 1.0  | 4:22.92  | Xorg            |
| 4676  | edwar | 20 | 0   | 22.9g   | 309772 | 91896  | S | 8.3  | 1.9  | 8:57.34  | Isolated Web Co |
| 9619  | edwar | 20 | 0   | 2013600 | 441608 | 152704 | S | 6.6  | 2.7  | 5:32.23  | soffice.bin     |
| 12094 | edwar | 20 | 0   | 636808  | 67308  | 44556  | S | 4.3  | 0.4  | 0:20.59  | gnome-terminal- |
| 4857  | edwar | 20 | 0   | 2898940 | 361264 | 97900  | S | 2.3  | 2.2  | 1:50.70  | Isolated Web Co |
| 24280 | edwar | 39 | 19  | 398576  | 35248  | 24208  | S | 2.0  | 0.2  | 0:00.06  | tracker-extract |
| 10561 | edwar | 20 | 0   | 773420  | 74156  | 46712  | S | 1.3  | 0.5  | 0:05.94  | gedit           |
| 4888  | edwar | 20 | 0   | 2861144 | 299876 | 94428  | S | 1.0  | 1.8  | 1:38.78  | Isolated Web Co |
| 4923  | edwar | 20 | 0   | 2841304 | 324336 | 97788  | S | 1.0  | 2.0  | 1:38.48  | Isolated Web Co |
| 24045 | edwar | 20 | 0   | 776044  | 75164  | 48948  | S | 1.0  | 0.5  | 0:01.90  | eog             |
| 2863  | edwar | 9  | -11 | 53880   | 31292  | 7556   | S | 0.7  | 0.2  | 0:00.65  | pipewire-pulse  |
| 3937  | edwar | 20 | 0   | 3071928 | 289552 | 39684  | S | 0.7  | 1.8  | 0:32.49  | dropbox         |
| 4704  | edwar | 20 | 0   | 2728580 | 304864 | 91676  | S | 0.7  | 1.9  | 0:57.74  | Isolated Web Co |
| 6237  | edwar | 20 | 0   | 1722768 | 177412 | 84944  | S | 0.7  | 1.1  | 0:34.49  | nautilus        |
| 2859  | edwar | 9  | -11 | 409364  | 20148  | 11816  | S | 0.3  | 0.1  | 0:00.52  | wireplumber     |
| 3121  | edwar | 20 | 0   | 12144   | 7424   | 3912   | S | 0.3  | 0.0  | 0:15.82  | dbus-daemon     |
| 3270  | edwar | 20 | 0   | 233364  | 7956   | 5272   | S | 0.3  | 0.0  | 0:00.05  | gvfs-mtp-volume |
| 4809  | edwar | 20 | 0   | 2637744 | 179692 | 86168  | S | 0.3  | 1.1  | 0:11.14  | Isolated Web Co |
| 5254  | edwar | 20 | 0   | 3074424 | 472256 | 92232  | S | 0.3  | 2.9  | 0:56.04  | Isolated Web Co |



## Comandos: **nice** e **renice**

- O comando **nice** é usado para inicializar um processo com uma determinada prioridade, exemplo:

```
nice -n 10 gedit
```

- O comando **renice** é usado para modificar a prioridade de um processo que já está em execução, exemplo:

```
ps -A | grep gedit
```

```
sudo renice -n -5 -p PID_do_GEdit
```

```
ps -elx
```



## Referências

- GUIA FOCA GNU/Linux. **Iniciante**. Disponível em: <http://www.guiafoca.org/cgs/guia/iniciante/ch-cmdc.html>. Acesso em: 06 ago. 2017.
- OPENAI. Esclarecimentos sobre os comandos nice e renice no Linux e elaboração de exercícios práticos (ChatGPT). São Paulo: OpenAI, 7 out. 2025. Disponível em: <https://chat.openai.com/>. Acesso em: 07 out. 2025.
- ROMERO, D. **Começando com Linux: comandos, serviços e administração**. [S. l.]: Casa do Código, 2014.