

# PROCESSOS

ADMINISTRAÇÃO DE SISTEMAS  
2022/2023

ROLANDO MARTINS  
(ADAPTADOS DE PEDRO BRANDÃO)

# Referências dos slides

---

- O conteúdo destes slides é baseado no livro da disciplina: “Unix and Linux System Administration Handbook (4ªEd)” por Evi Nemeth, Garth Snyder, Trent R. Hein e Ben Whaley, Prentice Hall, ISBN: 0-13-148005-7
- As imagens usadas têm a atribuição aos autores ou são de uso livre.

# Componentes de um processo

---

- Espaço de endereçamento
- Estado (*sleeping, stopped, runnable, etc.*)
- Prioridade de execução
- Informação sobre os recursos utilizados
- Informação sobre ficheiros e portas de rede abertos
- Máscara de sinais (registo dos sinais bloqueados)
- Dono do processo

# Atributos do processo

---

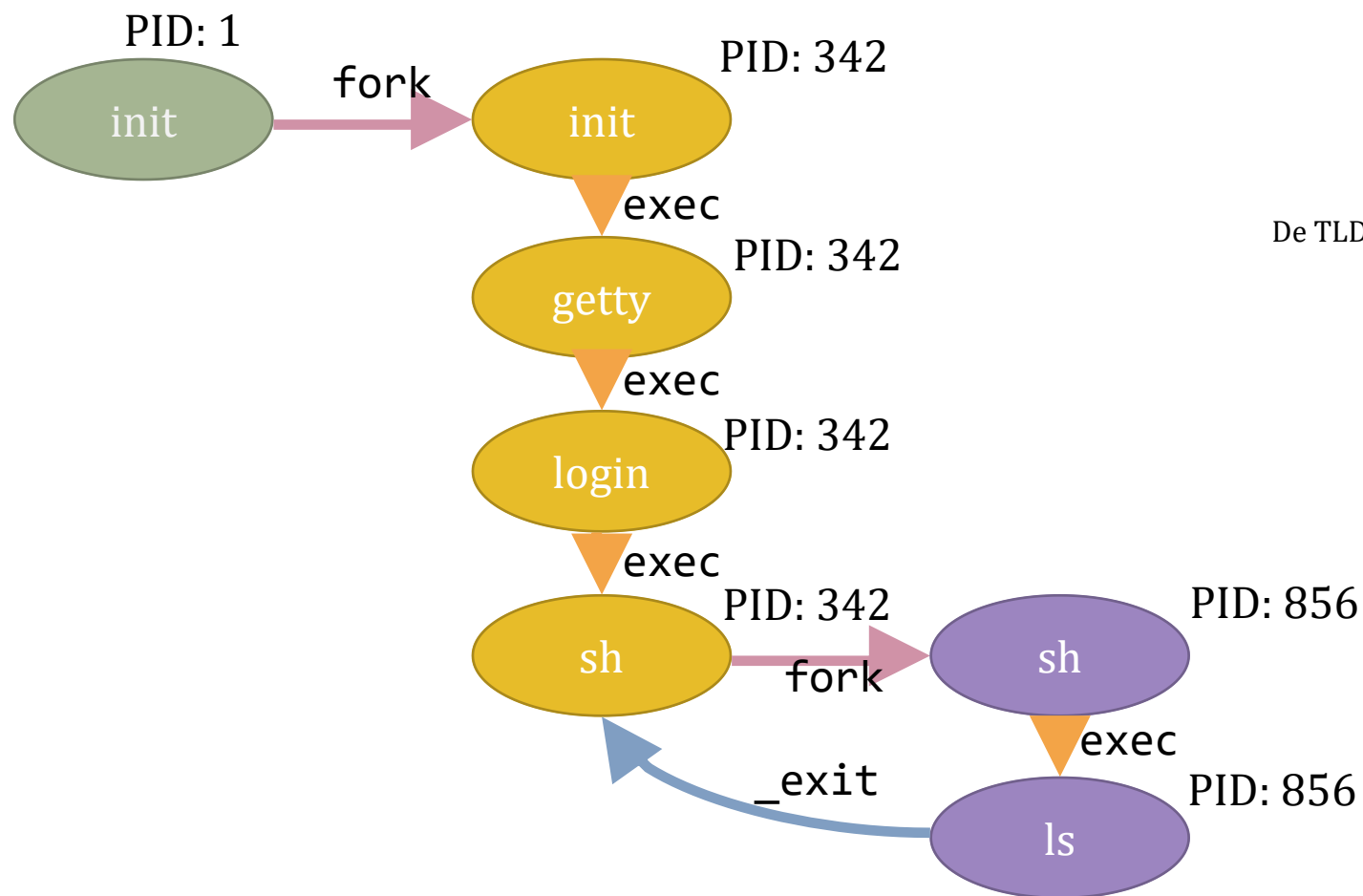
- PID – Process ID
  - Número único atribuído pelo kernel
    - Há *namespaces* que permitem o mesmo PID a diferentes processos
- PPID – Parent PID
  - Processo que se clonou para criar o processo corrente
- UID e EUID – real and effective user ID
  - ID do utilizador que criou o processo (cópia do UID do pai)
  - EUID pode ser != UID, exemplo `setuid`
- GID e EGID – real and effective group ID

# Atributos do processo (2)

---

- Niceness
  - Relacionado com a prioridade
  - Determina o escalonamento para ser processado
- Terminal de controlo
  - `stdin`, `stdout`, `stderr`
  - sinais

# Ciclo de vida



De TLDP – [Processes Inside Out](#)

# SINAIS

# Sinais

---

- Meio de comunicação entre processos
- Terminar, interromper ou suspender um processo pelo terminal
- Enviados pelo kernel a processos que cometem infrações (ex.: divisão por zero)
- Avisos do kernel ao processo (morte de um filho, disponibilidade de um canal de E/S)
- Etc.



# Tratamento

---

- Processo tem um *handler* para o sinal
- ou
- Kernel tem tratamento por omissão



# Alguns sinais

#	Nome	Descrição	Por omissão	Pode ser apanhado?	Pode ser bloqueado?	Coredump?
1	HUP	Hangup	Terminate	Yes	Yes	No
2	INT	Interrupt	Terminate	Yes	Yes	No
3	QUIT	Quit	Terminate	Yes	Yes	Yes
9	KILL	Kill	Terminate	No	No	No
*	BUS	Bus error	Terminate	Yes	Yes	Yes
11	SEGV	Segmentation fault	Terminate	Yes	Yes	Yes
15	TERM	Software termination	Terminate	Yes	Yes	No
*	STOP	Stop	Stop	No	No	No
*	TSTP	Keyboard stop	Stop	Yes	Yes	No
*	CONT	Continue after stop	Ignore	Yes	No	No
*	WINCH	Window changed	Ignore	Yes	Yes	No
*	USR1	User-defined #1	Terminate	Yes	Yes	No
*	USR2	User-defined #2	Terminate	Yes	Yes	No

- Lista disponível com

`kill -l`

- \* Varia de acordo com o sistema

# Envio de sinais



## kill - terminate a process

- Em geral:

```
kill [-signal] pid
```

Por omissão o sinal é TERM

## killall - kill processes by name

- Em Linux define processo(s) por nome

```
killall emacs
```

- Em UNIX termina todos os processos do utilizador

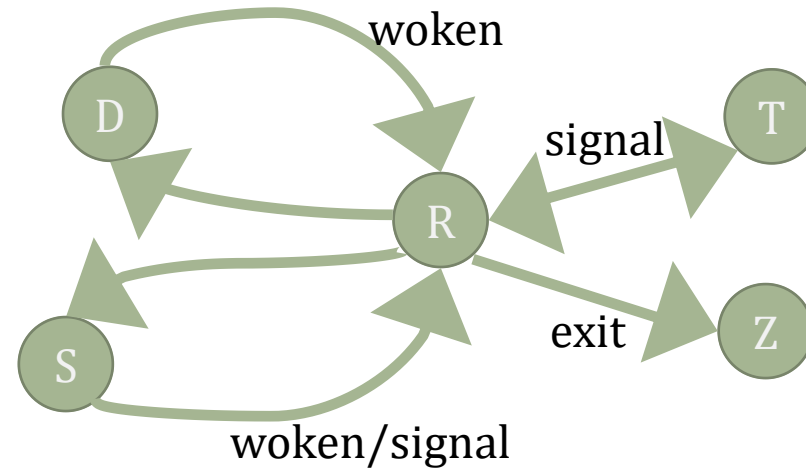
- Incluindo init se o utilizador for root

## pgrep, pkill - look up or signal processes based on name and other attributes

- Listar ou enviar sinal para processos de acordo com opções

# GESTÃO DE PROCESSOS

# Estados



Baseado em [process states](#)

`man ps`

- D uninterruptible sleep (usually IO)
- R running or runnable (on run queue)
- S interruptible sleep (waiting for an event to complete)
- T stopped by job control signal
- ...
- Z defunct ("zombie") process, terminated but not reaped by its parent

# Prioridade

---

- Valor nice associado ao processo:
  - De 19 (o mais simpático) a -20 (nada simpático para os outros) (em Linux/UNIX)
  - Ver no livro outros
- Serve para o kernel escalonar os processos
- Processo herda valor do pai
- Pode ser aumentado pelo utilizador, mas não diminuído
  - Superuser pode tudo...

# Prioridade - exemplos

---

```
$ nice -n 5 ~/bin/longtask
```

```
// launch process with lower priority (raise nice) by 5
```

```
$ sudo renice -5 8829
```

```
// Sets nice value to -5
```

```
$ sudo renice 5 -u boggs
```

```
// Sets nice value of boggs's procs to 5
```

# ps

---

- Informações sobre os processos correntes do sistema
- Vários parâmetros, mais usados

`ps aux # BSD style options`

- ou

`ps -ef # standard options`



# top

---

- Mostra dados semelhantes ao ps, mas continuamente
- Tem flags para o formato e o que mostrar como o ps
- Com cores e interação

htop

# /proc

---

- Pseudo sistema de ficheiros (man proc)
- Vários dados sobre o sistema
- Leitura
  - `cat /proc/meminfo`
  - `cat /proc/cpuinfo`
- Dados sobre os processos (/proc/<PID>/)
- Escrita para alteração de parâmetros
  - `echo 1 > /proc/sys/net/ipv4/ip_forward`
    - Colocar o kernel a fazer routing IP

# strace

---

- Dá informação sobre as chamadas de sistema do processo

`strace -p 9180`

- Para saber os processos

- Ver processos do utilizador

`ps -au auser`

- Ver o top

# uptime

---

- Dá utilização do sistema em processos
- Em intervalos de 1, 5 e 15 min

```
$ uptime
```

```
18:54:27 up 19:28, 5 users, load average: 0.01, 0.04, 0.05
```

- Permite ver a carga do sistema e tentar com as outras ferramentas descobrir “a culpa” de algum excesso de carga

# resumo

---

- Componentes e atributos de um processo
- Ciclo de vida
- Sinais
- Gestão de processos

# QUESTÕES/ COMENTÁRIOS