

**INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA**  
RIO GRANDE DO NORTE

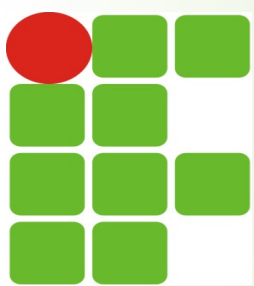


**REDE FEDERAL  
DE EDUCAÇÃO  
PROFISSIONAL  
E TECNOLÓGICA**  
1909-2009

# Sistemas Operacionais de Redes

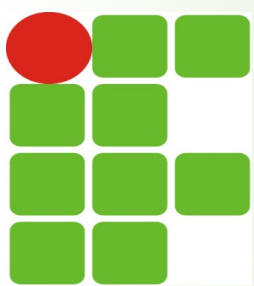
Aula: Gerenciamento de processos

Professor: Jefferson Silva



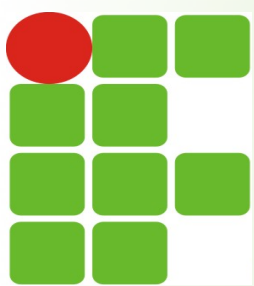
# Agenda

- Conhecer o que é um processo
- Quais as funções de um processo?
- Gerenciar processos
- Finalizar processos
- Referências



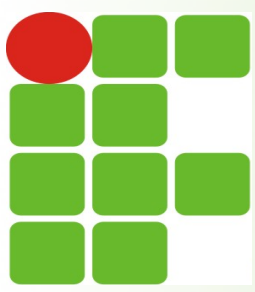
# O que é um processo ?

- Um processo é basicamente um programa em execução (TANENBAUM, 2014)
- Um processo é o contêiner que detém todas as informações necessárias para rodar um programa (TANENBAUM, 2014)
  - Cada programa detém o seu processo

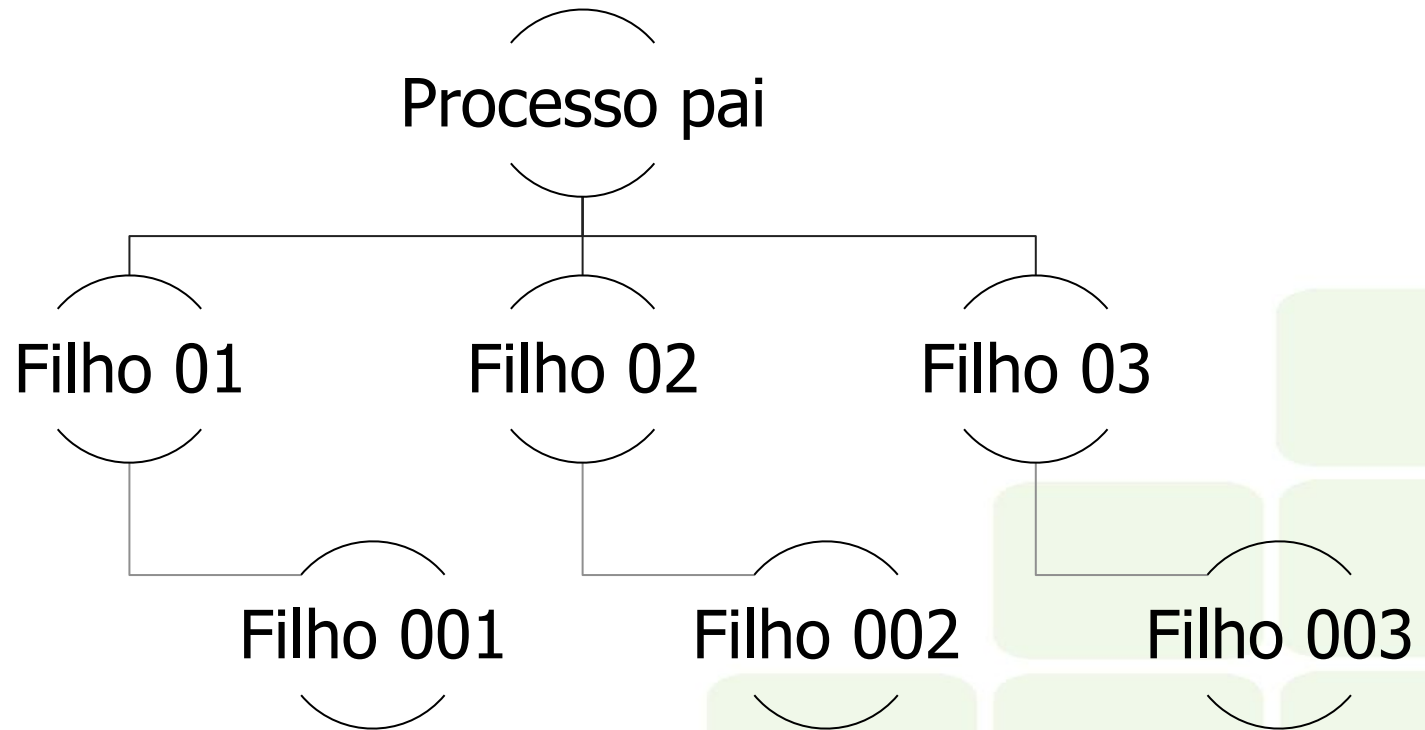


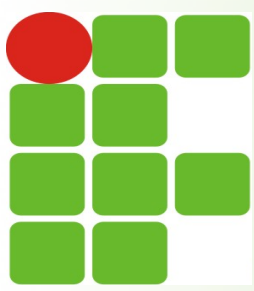
# O que é um processo ?

- Um processo pode ter processos filhos
- Cabe ao sistema operacional gerenciar os processos correntes



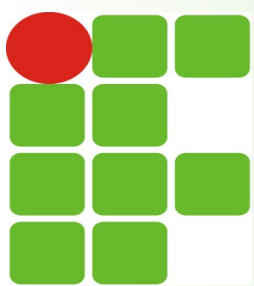
# Hierarquia de processos





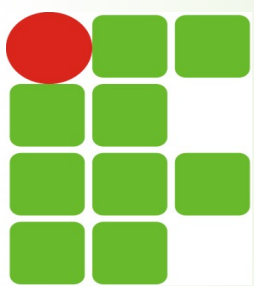
# Exemplo de hierarquia - Linux

```
root@jarvis:~# pstree
init--VBoxService--7*[{VBoxService}]
      --acpi_fakekeyd
      --acpid
      --atd
      --avahi-daemon--avahi-daemon
      --bluetoothd
      --console-kit-dae--64*[{console-kit-dae}]
      --cron
      --cupsd
      --dbus-daemon
      --dhclient
      --exim4
      --5*[getty]
      --login--bash--top
      --polkitd--{polkitd}
      --rpc.idmapd
      --rpc.statd
      --rpcbind
      --rsyslogd--3*[{rsyslogd}]
      --sshd--sshd--bash--pstree
      --udevd--2*[udevd]
```



# Exemplo de hierarquia - Linux

```
jefferson@monaco:~$ pstree
init--/opt/zimbra/lib--zmlogger--zmlogger: zmrtd
  |
  |--acpid
  |--atd
  |--atop
  |--cron
  |--dbus-daemon
  |--6*[getty]
  |--httpd--3*[httpd--26*[{httpd}]]
  |         |--2*[rotatelogs]
  |--irqbalance
  |--java--19*[{java}]
  |--master--anvil
  |          |--cleanup
  |          |--lmtop
  |          |--pickup
  |          |--2*[proxymap]
  |          |--qmgr
  |          |--showq
  |          |--2*[smtp]
  |          |--3*[smtpd]
  |          |--tlsmgr
  |          |--trivial-rewrite
  |--memcached--5*[{memcached}]
  |--mysqld_safe--mysqld--22*[{mysqld}]
  |--opendkim--opendkim--2*[{opendkim}]
```



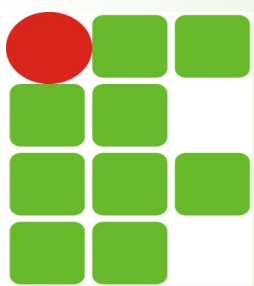
# **COMO GERENCIÁ-LOS ?**

# **O QUE GERENCIAR ?**

22/09/14

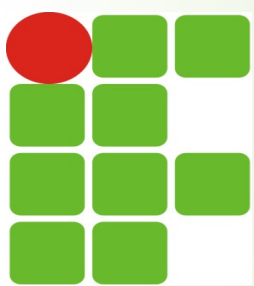
Instituto Federal de Educação,  
Ciência e Tecnologia do RN - IFRN





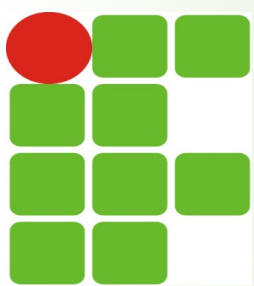
# Gerenciamento de processos

- Cabe ao Sistema Operacional (SO) gerenciar todos os recursos da máquina
- Os processos demandam recursos (que são finitos)

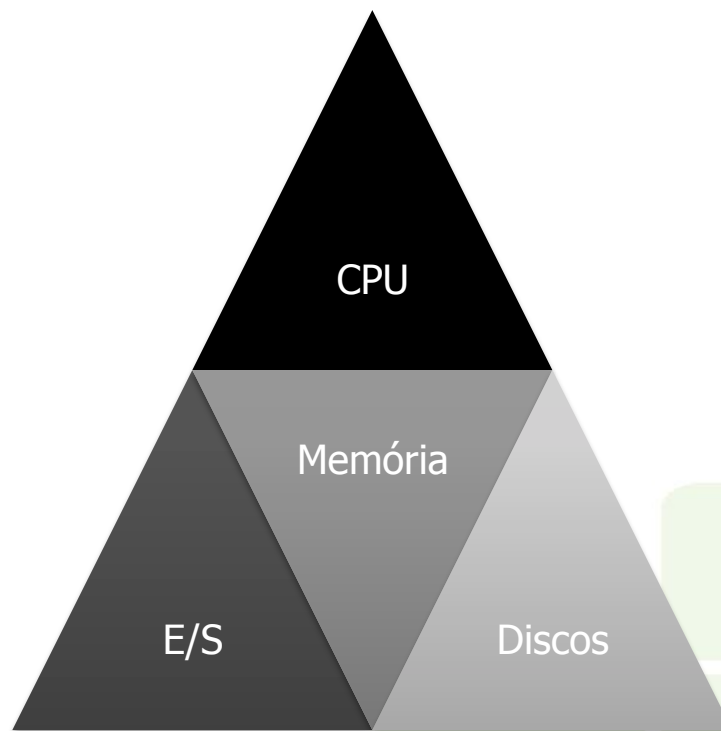


# Gerenciamento de processos

- O SO precisa conceder os recursos, mas “vigia-los”
- Apenas alguns processos são executados por vez

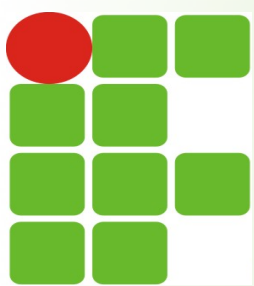


# Gerenciamento de recursos



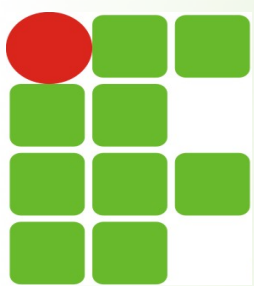
22/09/14

Instituto Federal de Educação,  
Ciência e Tecnologia do RN - IFRN



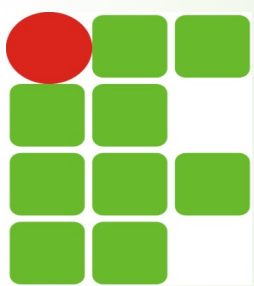
# Como o SO gerencia os processos ?

- Por meio das prioridades
  - Cada processo tem um *Process ID* (PID)
  - Cada PID tem a sua prioridade
- No Linux, as prioridades vão de -20 a 19
  - Quanto menor, maior a prioridade



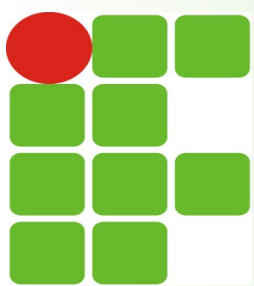
# Gerenciando processos no Linux

- Quando o computador é ligado o BIOS procura pelo gerenciador de boot no HD
  - Os gerenciadores mais comuns são o GRUB e o LILO
- O GRUB ou LILO carregam o kernel na memória
- O kernel inicia um processo chamado init
- O init carrega todas as outras aplicações
- Por fim, o init inicia o getty – é o responsável pela autenticação dos usuários e inicia o processo do shell



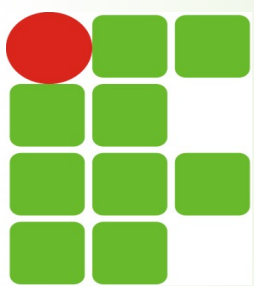
# Exemplo

```
root@jarvis:~# pstree
init--VBoxService--7*[{VBoxService}]
    --acpi_fakekeyd
    --acpid
    --atd
    --avahi-daemon--avahi-daemon
    --bluetoothd
    --console-kit-dae--64*[{console-kit-dae}]
    --cron
    --cupsd
    --dbus-daemon
    --dhclient
    --exim4
    --5*[getty]
    --login--bash--top
    --polkitd--{polkitd}
    --rpc.idmapd
    --rpc.statd
    --rpcbind
    --rsyslogd--3*[{rsyslogd}]
    --sshd--sshd--bash--pstree
    --udevd--2*[udevd]
```



# Como gerenciar efetivamente ?

- Os comandos que iremos utilizar são:
  - `ps [opções]`
    - Gera uma lista com todos os processos em execução
  - `pstree [opções]`
    - Mostra os processos na hierarquia
  - `top [opções]`
    - Mostra os processos em execução (como o `ps`), mas atualizando a tela



# Como gerenciar efetivamente ?

## ■ Os comandos que iremos utilizar são:

- `kill [opções] PID`

- O seu principal uso é para matar (via PID) os processos em execução

- `killall [opções] nome`

- Similar ao kill, mas mata pelo nome

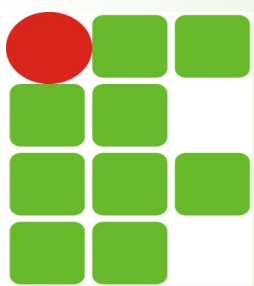
- `nohup [comando]`

- Executa uma aplicação e a mantém funcionando independente do usuário estar conectado ou não

- `free [opções]`

- Mostra a quantidade de memória disponível no sistema





# Alterando a prioridade dos processos

- Nice [*prioridade*] comando
  - “legal”, em inglês. Usado para definir a prioridade de um programa que será iniciado
  - Quanto “menos legal” com os outros, maior a sua prioridade
- Renice [+/- *prioridade*] [*opções*] PID/  
usuario
  - Utilizado para alterar a prioridade de um processo já em execução