

Gerenciamento de memória e processamento

Comandos de Gerência de Memória

free

O Comando **free** mostra espaços livres e ocupados em memória virtual (RAM+Swap).

Sintaxe:

```
#free
```

Opções:

- m - mostra resultados em MB.
- k - mostra resultados em KB.
- t - exibe uma lista denominada total, onde consta a memória virtual.

memstat

O Comando **memstat** mostra o uso da memória por parte dos diferentes processos. Útil para descobrir qual processo está consumindo exageradamente a memória.

Sugestão de uso para saber quem mais consome RAM:

Sintaxe:

```
#memstat | sort -nr | head -n 20
```

pmap

Mostra, detalhadamente , como um processo está utilizando a memória, incluindo outros processos dos quais ele dependa.

Sintaxe:

#pmap [processo]

Comandos de Gerencia de Processamento

ps

O comando **ps** mostra os processos que estão ativos.

Sintaxe:

ps [parâmetros]

Opções:

a – Mostra os processos qu rodam em terminais, independentemente de qual usuário seja o dono de tal processo.

f - Mostra os processos filhos ads– (threads) ligados aos seus processos-pai.

x -Mostra os processos, cujo o dono seja o mesmo usuário que emitiu o programa

Comandos de Gerencia de Processamento

-A – o mesmo que ax.

u - Mostra os dados de forma orientada a usuários, ou seja, adicionando colunas que indicam quem é o usuário dono do processo, o consumo de CPU filhos ads– (threads) ligados aos seus processos-pai.

x -Mostra os processos, cujo o dono seja o mesmo usuário que emitiu o programa

Comandos de Gerencia de Processamento

kill

Envia um sinal para o processo em execução. Geralmente é utilizado para encerrarmos processos. Ele pode enviar vários tipos de sinais para os processos. Esses sinais poderão ser vistos no comando (kill -l).

Sinais Importantes:

Chave	Função
-------	--------

-9	Mata o processo não admitindo bloqueios por parte do sistema.
----	---

-kill	o mesmo que -9.
-------	-----------------

-l	Reinicia o processo.
----	----------------------

-HUP	O mesmo que -l
------	----------------

Comandos de Gerencia de Processamento

Exemplos do uso do **kill**

Comando Resultado

#kill 2021 Mata o processo com o PID 2021. O sistema poderá bloquear esta operação caso não seja oportuno o encerramento do processo.

#kill -9 2021 Mata o processo com o PID 2021, mesmo que o sistema tente bloquear esta ação.

#kill -HUP 2021 Reinicializa o processo número 2021.

Comandos de Gerencia de Processamento

Killall (instale com `#apt-get install psmisc`)

O **killall** é similar ao **kill**, a diferença é que no **killall** podemos usar o nome do processo em vez do PID.

Exemplos:

Comando	Resultado
---------	-----------

<code>#Killall mcedit</code>	Mata todos os processos mcedit que estiverem em execução.
------------------------------	---

<code>#killall -r `.*aleo.*`</code>	Mata todos os processos cujo os nomes conheçam a expressão aleo.
-------------------------------------	--

Comandos de Gerencia de Processamento

Jobs

O comando Jobs, mostra os processos que está sendo executado em segundo plano.

Sintaxe:

#jobs

Comandos de Gerencia de Processamento

bg

O comando **bg**, reinicia o backgroud (segundo plano), a execução de um processo parado por Ctrl z. **Um programa é executado em segundo plano, quando não ocupa um terminal.**

Exemplo:

Comando Resultado

#bg 3 Reinicia o processo número 3 em backgroud.
O Número 3 foi obtido a partir do comando jobs

Comandos de Gerencia de Processamento

fg

O comando **fg**, reinicia o foreground(primeiro plano), a execução de um processo parado por Ctrl z. **Um programa é executado em primeiro plano, quando ocupa um terminal.**

Exemplo:

Comando	Resultado
---------	-----------

#fg 2	Reinicia o processo número 2 em foreground.
-------	---

O Número 2 foi obtido a partir do comando jobs

Comandos de Gerencia de Processamento

time

O comando **time**, mede o tempo de execução de um programa. o foreground(primeiro plano), a execução de um processo parado por Ctrl z. **Um programa é executado em primeiro plano, quando ocupa um terminal.**

Sintaxe:

```
#time updatedb
```

“mede o tempo de execução do comando updatedb

Expressão Significado

user refere-se ao tempo de utilização da CPU gasto pelo comando time.

sys refere-se ao tempo que o sistema operacional ocupou de CPU para fazer alocações necessárias.

Comandos de Gerencia de Processamento

nohup

Permite que um comando continua a ser executado, mesmo que o usuário faça logout do sistema

Exemplo:

Comando	Resultado
#nohup updatedb & updatedb	Executa em segundo plano por causa do caractere &, o comando

Comandos de Gerencia de Processamento

strace (instale com o #apt-get install strace)

Monitora todas as chamadas e sinais que um processo envia para o sistema. É excelente para encontrar erros e problemas em processos em tempo de execução.

Exemplo:

Comando	Resultado
#nohup updatedb & updatedb	Executa em segundo plano por causa do caractere &, o comando

Comandos de Gerencia de Processamento

fuser (instale com o #apt-get install psmisc)

O comando fuser serve para mostrar quais processos estão utilizando determinado filesystem, arquivo, diretório ou elemento de rede, como uma porta local ou remota.

Exemplo:

Comando	Resultado
---------	-----------

#fuser /	
----------	--

Retorno: 2275r 2321rc	
-----------------------	--

Comandos de Gerencia de Processamento

Letra	Significado
c	Diretório atual
e	Arquivo em execução
r	Diretório arquivo em memória
m	Arquivo em memória
f	Arquivo aberto para leitura
F	Arquivo aberto para escrita

Comandos de Gerencia de Processamento

Isof (instale com o #apt-get install psmisc)

O comando **Isof** mostra quais arquivos estão abertos em RAM por processos.

Chave	Função
-------	--------

- | | |
|--------|--|
| -p PID | Mostra os arquivos que possuem ligação com o PID mostrado. |
| -u UID | Mostra os processos sob a gestão do usuário em questão. |
| -i | Mostra os processos relacionados as portas TCP ou UDP abertas. |