

Comandos de memória no Linx

1. Comando bpytop

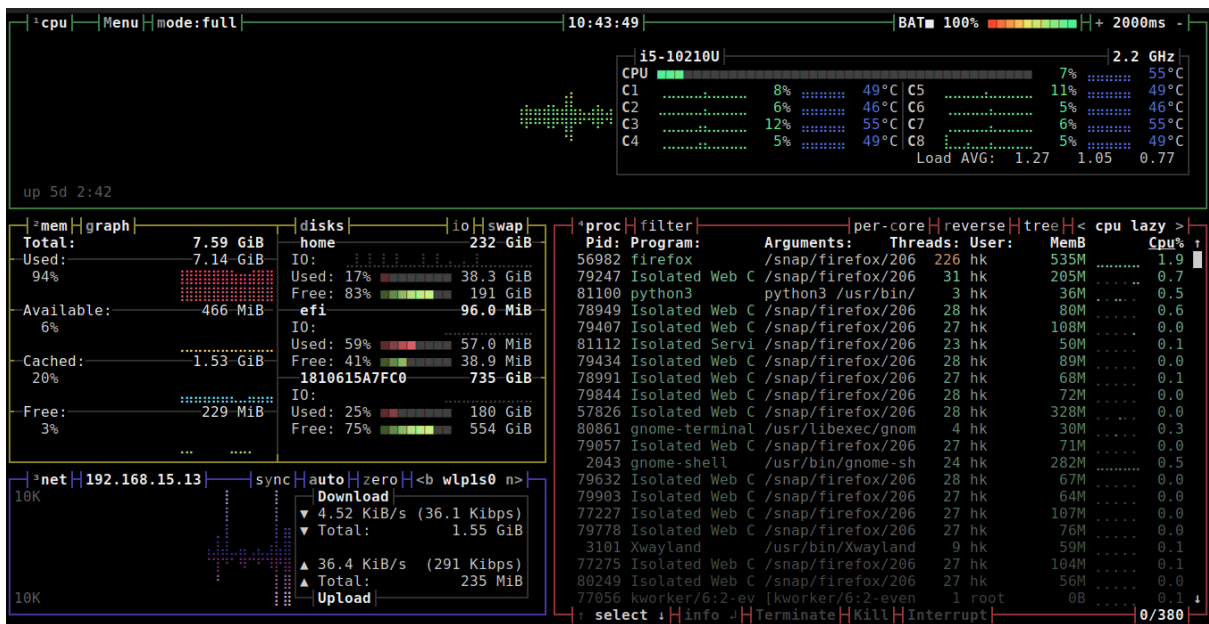
O bpytop é um comando que exibe uma interface gráfica de monitoramento de processos, memória e CPU desenvolvido em Python, no entanto, por não ser nativa do Linux como o “top”, assim como o “htop” necessita ser instalado.

Instalação:

```
$ sudo apt install bpytop
```

Uso:

```
$ bpytop
```



Para alterar configurações do bpytop clicar em “esc” e a seguir, com a seta, seleccionar “options”, ou ajuda, com a seta seleccionar “help”. ou sair, com a seta seleccionar “quit”.



2. Comando free

O free é um comando para verificar o uso da memória no linux, que por padrão exibe a saída de dados em KB.

Exemplo:

```
$ free
```

	total	used	free	shared	buff/cache	available
Mem:	7966116	6115736	457016	948668	1393364	634024
Swap:	0	0	0			

A opção -m exibe todos os dados em MB:

```
$ free -m
```

	total	used	free	shared	buff/cache	available
Mem:	7779	5975	433	920	1370	625
Swap:	0	0	0			

Assim como a opção -g exibe todos os dados em GB:

```
$ free -g
```

	total	used	free	shared	buff/cache	available
Mem:	7	5	0	0	1	0
Swap:	0	0	0			

Um detalhe interessante é que na coluna "buff/cache" cache e buffers aproximadamente 1370 MB estão sendo utilizados, isso ocorre porque o Linux, por padrão, armazena muitos dados em cache para obter um desempenho mais rápido, para que a memória possa ser liberada e usada, se necessário.

As colunas na saída do free significam:

- total: representa a quantidade total de memória que pode ser usada pelos aplicativos.

- used: é a memória usada e é calculado da seguinte forma:

$$\text{used} = \text{total} - \text{free} - \text{buffers} - \text{cache}$$

- free: memória livre / não utilizada.

- shared: a coluna pode ser ignorada porque não tem significado e está visível apenas para compatibilidade com versões anteriores do Linux.

- buff/cache: representa a memória combinada usada pelos buffers do kernel e cache de página e blocos, que pode ser recuperada a qualquer momento utilizando ferramentas específicas. Para exibir os buffers e o cache em duas colunas separadas usar a opção -w .

- available: estima a quantidade de memória disponível para iniciar novos aplicativos, sem troca, ou seja, sem precisar usar swap.

A opção -h exibe as informações com as unidades.

Exemplo:

```
$ free -h
```

	total	used	free	shared	buff/cache	available
Mem:	7,6Gi	4,2Gi	208Mi	1,8Gi	3,2Gi	1,4Gi
Swap:	0B	0B	0B			

3. Arquivo /proc/meminfo

Outra forma para verificar o uso da memória é usar o conteúdo do arquivo /proc/meminfo. A pasta/diretório /proc não contém arquivos reais já que só é criado quando o sistema operacional é executado e seu conteúdo é montado, portanto, só existe enquanto o sistema operacional está executando.

O /proc é constituído por arquivos virtuais que contêm todas as informações dinâmicas sobre o kernel e o sistema em geral.

Exemplo:

```
$ cat /proc/meminfo
```

```
MemTotal: 7966116 kB  
MemFree: 173708 kB  
MemAvailable: 1365296 kB  
Buffers: 8 kB  
Cached: 3233980 kB  
SwapCached: 0 kB  
SwapTotal: 0 kB  
SwapFree: 0 kB  
Dirty: 628 kB  
Writeback: 0 kB  
AnonPages: 4021512 kB  
Mapped: 657852 kB  
Shmem: 1864804 kB  
KReclaimable: 102516 kB  
Slab: 279944 kB  
SReclaimable: 102516 kB  
SUnreclaim: 177428 kB  
KernelStack: 23520 kB  
PageTables: 64988 kB  
NFS_Unstable: 0 kB  
Bounce: 0 kB  
WritebackTmp: 0 kB  
CommitLimit: 3983056 kB  
Committed_AS: 14955100 kB  
VmallocTotal: 34359738367 kB  
VmallocUsed: 64656 kB  
VmallocChunk: 0 kB  
Percpu: 10208 kB  
HardwareCorrupted: 0 kB  
AnonHugePages: 0 kB  
ShmemHugePages: 0 kB  
ShmemPmdMapped: 0 kB  
FileHugePages: 0 kB  
FilePmdMapped: 0 kB  
HugePages_Total: 0  
HugePages_Free: 0  
HugePages_Rsvd: 0  
HugePages_Surp: 0  
Hugepagesize: 2048 kB
```

Hugetlb: 0 kB
DirectMap4k: 1138108 kB
DirectMap2M: 7077888 kB
DirectMap1G: 1048576 kB

Informações no meminfo:

- MemTotal: memória RAM total utilizável, calculado da seguinte forma:

MemTotal = RAM_total - bits reservados pelo SO - código binário do kernel

- MemFree: soma de LowFree + HighFree

- MemShared : aparece por razões históricas, é sempre 0 (zero).

- Buffers: memória no cache do buffer. É inútil como métrica nos sistemas Linux atuais, pois o armazenamento relativamente temporário para blocos de disco não deve ser grande (20 MB ou mais);

- Em cache: memória no pagecache (diskcache) - SwapCache (não inclui SwapCached)

- SwapCache : memória que já foi trocada, mas ainda está no arquivo de troca (se esta porção de memória for solicitadas para uso, estará disponível para ser sobrescrito, pois há economia de E/S;

Estatísticas da memória

- HighTotal: é a quantidade total de memória na região alta. Highmem é toda a memória acima de (aproximadamente) 860 MB de RAM física. Kernel usa truques indiretos para acessar a região de alta memória. O cache de dados pode ficar nesta região da memória.

- LowTotal: a quantidade total de memória não alta.

- LowFree: A quantidade de memória livre da região de pouca memória. Esta é a memória que o kernel pode endereçar diretamente. Todas as estruturas de dados do kernel precisam ficar com pouca memória.

- SwapTotal: Quantidade total de memória swap física.

- SwapFree: Quantidade total de memória swap livre. Memória que foi removida da RAM e está temporariamente no disco

- Suja: memória que está esperando para ser gravada de volta no disco

- Writeback: Memória que está ativamente sendo gravada de volta no disco

- Mapeados: arquivos que foram mapeados, como bibliotecas

- Slab: cache de estruturas de dados no kernel

- Committed_AS: uma estimativa de quanta RAM você precisaria para ter uma garantia de 99,99% de que nunca haverá OOM (sem memória) para esta carga de trabalho. Normalmente, o kernel compromete a memória. Isso significa que, digamos que você faça um malloc de 1 GB, nada acontece, realmente. Somente quando você começar a USAR aquela memória malloc, você obterá memória real sob demanda, e tanto quanto você usa. Então você meio que faz uma hipoteca e espera que o banco não vá à falência. Outros casos podem incluir quando você faz o mapeamento de um arquivo que é compartilhado apenas quando você grava nele e obtém uma cópia privada desses dados. Embora normalmente seja compartilhado entre os processos. O Committed_AS é uma estimativa de quanta RAM / swap você precisaria no pior caso.

- PageTables: quantidade de memória dedicada ao nível mais baixo de tabelas de páginas.

- ReverseMaps: número de mapeamentos reversos realizados

- VmallocTotal: tamanho total da área de memória vmalloc

- VmallocUsed: quantidade de área vmalloc que é usada

- VmallocChunk: o maior bloco contíguo da área de vmalloc que é gratuito

Estatísticas da VM (Virtual Memory - Swap)

A VM divide as páginas de cache em memória “ativa” e “inativa”, portanto, a ideia é de que se o SO precisa de memória e algum cache precisa ser sacrificado para isso, o SO pode retirá-lo do status inativo, pois espera-se que ele não seja usado, dessa forma, o SO verifica o que é usado regularmente e movimenta as coisas.

- Ativo: quando a memória que foi usada mais recentemente e geralmente não recuperada, a menos que seja absolutamente necessário.

- Inact_dirty: neste caso, dirty significa “pode ser necessário gravar no disco ou trocar”. Dá mais trabalho para liberar. Os exemplos podem ser arquivos que ainda não foram gravados. Eles não são gravados na memória muito cedo para manter o I / O baixo. Por exemplo, se você estiver gravando logs, pode ser melhor esperar até ter um log completo pronto antes de enviá-lo para o disco.

- Inact_clean: presume-se que seja facilmente liberável, assim, o kernel tentará manter algumas coisas limpas sempre para ter um pouco de espaço para respirar.

- Inact_target: apenas uma métrica de meta que o kernel usa para garantir que haja páginas inativas suficientes ao redor. Quando excedido, o kernel não funcionará para mover as páginas de ativas para inativas. Uma página também pode ficar inativa de algumas outras maneiras, por exemplo, se você fizer uma longa E / S sequencial, o kernel assume que você não vai usar essa memória e a torna inativa preventivamente. Portanto, você pode obter mais páginas inativas do que o destino porque o kernel marca algum cache como “mais provável de nunca ser usado” e permite que ele trapaceie na ordem de “última utilização”.

4. Comando vmstat

O comando vmstat apresenta as estatísticas de uso da memória.

Utilizando a opção -s, a saída é exibida como no comando proc.

Exemplo:

```
$ vmstat
```

```
procs -----memory----- ---swap-- -----io----- -system-- -----cpu-----
r b swpd free buff cache si so          bi    bo in  cs us sy id wa st
1 0  0 320624    8 3121252    0    0  1112  255 254 124 30 11 58 0 0
```

```
$ vmstat -s
```

```
7966116 K total memory
4524696 K used memory
572296 K active memory
4999844 K inactive memory
319168 K free memory
8 K buffer memory
3122244 K swap cache
0 K total swap
0 K used swap
0 K free swap
3836546 non-nice user cpu ticks
9936 nice user cpu ticks
1368329 system cpu ticks
7388623 idle cpu ticks
10261 IO-wait cpu ticks
0 IRQ cpu ticks
```

```
70994 softirq cpu ticks
0 stolen cpu ticks
141041695 pages paged in
32293977 pages paged out
0 pages swapped in
0 pages swapped out
246932763 interrupts
531112331 CPU context switches
1668168107 boot time
85511 forks
```

Opções para o comando Linux vmstat:

- a: Mostra memória ativa e inativa
- f: Mostra o número de forks desde a última reinicialização do sistema
- m: Mostra slabinfos
- s: Mostra estatísticas de memória e contadores
- d: Mostra estatísticas relacionadas ao disco.

5. Commando dmidecode

Para informações de hardware da RAM instalada, pode-se usar o comando demidecode, que exibe todos os detalhes sobre os dispositivos de memória RAM.

Comando genérico sobre os dispositivos de memória:

```
$ sudo dmidecode -t 16
# dmidecode 3.3
Getting SMBIOS data from sysfs.
SMBIOS 3.2.0 present.
Handle 0x000D, DMI type 16, 23 bytes
Physical Memory Array
  Location: System Board Or Motherboard
  Use: System Memory
  Error Correction Type: None
  Maximum Capacity: 8 GB
  Error Information Handle: No Error
  Number Of Devices: 2
```

Comando com mais especificações sobre a memória:

```
$ sudo dmidecode -t 16
# dmidecode 3.3
Getting SMBIOS data from sysfs.
SMBIOS 3.2.0 present.

Handle 0x000D, DMI type 16, 23 bytes
Physical Memory Array
  Location: System Board Or Motherboard
  Use: System Memory
  Error Correction Type: None
  Maximum Capacity: 8 GB
```

Error Information Handle: No Error
Number Of Devices: 2

...

Comando completo com informações sobre a(s) memória(s):

```
$ sudo dmidecode
```

```
[sudo] password for hk:
```

```
# dmidecode 3.3
```

```
Getting SMBIOS data from sysfs.
```

```
SMBIOS 3.2.0 present.
```

```
Table at 0x97DE0000.
```

Handle 0x0000, DMI type 0, 26 bytes

BIOS Information

Vendor: Insyde Corp.

Version: V1.10

Release Date: 06/10/2020

Address: 0xE0000

Runtime Size: 128 kB

ROM Size: 0 MB

Characteristics:

PCI is supported

BIOS is upgradeable

BIOS shadowing is allowed

Boot from CD is supported

Selectable boot is supported

EDD is supported

...

Commando para ver informações específicas sobre a memória RAM e seu hardware:

```
$ sudo dmidecode --type 17
```

```
Getting SMBIOS data from sysfs.
```

```
SMBIOS 3.2.0 present.
```

Handle 0x000E, DMI type 17, 84 bytes

Memory Device

Array Handle: 0x000D

Error Information Handle: No Error

Total Width: 64 bits

Data Width: 64 bits

Size: 4 GB

Form Factor: SODIMM

Set: None

Locator: ChannelA-DIMM0

Bank Locator: BANK 0

Type: DDR4

Type Detail: Synchronous

Speed: 2667 MT/s

...

Handle 0x000F, DMI type 17, 84 bytes

Memory Device

Array Handle: 0x000D

Error Information Handle: No Error

Total Width: 64 bits

Data Width: 64 bits

Size: 4 GB

Form Factor: SODIMM

Set: None

Locator: ChannelB-DIMM0

Bank Locator: BANK 2

Type: DDR4

Type Detail: Synchronous

Speed: 2667 MT/s

...

6. Comando lshw

o lshw (list hardware) informações sobre todo o hardware do computador e de forma bastante completa, no entanto, pode ser utilizado, com opções específicas, para exibir informações sobre a memória RAM do computador.

Obs: recomenda-se que o comando seja executado como usuário administrador.

Comando para exibir informações de todo o hardware do computador.

```
$ sudo lshw
```

```
hk
```

```
description: Notebook
```

```
product: Aspire A315-54 (0000000000000000)
```

```
vendor: Acer
```

```
version: V1.10
```

```
serial: NXHU3AL00503316F659501
```

```
width: 64 bits
```

```
capabilities: smbios-3.2.0 dmi-3.2.0 smp vsyscall32
```

```
configuration: chassis=notebook family=Aspire 3 sku=0000000000000000
```

```
uuid=ce5eecda-22d7-ea11-a41b-7c8ae1db9c93
```

```
*-core
```

```
description: Motherboard
```

```
product: Sleepy_WC
```

```
vendor: CML
```

```
physical id: 0
```

```
version: V1.10
```

```
serial: NBHQ211002034016D99501
```

```
slot: Type2 - Board Chassis Location
```

```
*-firmware
```

```
description: BIOS
```

```
vendor: Insyde Corp.
```

```
physical id: 0
```

```
version: V1.10
```


date: 06/10/2020
size: 128KiB

...

Comando lshw com descrições de memória, buffers e caches:

\$ sudo lshw -C memory

*-firmware

description: BIOS
vendor: Insyde Corp.
physical id: 0
version: V1.10
date: 06/10/2020
size: 128KiB

capabilities: pci upgrade shadowing cdboot bootselect edd int13floppy nec
int13floppy toshiba int13floppy360 int13floppy1200 int13floppy720 int13floppy2880
int9keyboard int10video acpi usb biosboot specification uefi

*-cache:0

description: L1 cache
physical id: 5
slot: L1 Cache
size: 256KiB
capacity: 256KiB
capabilities: synchronous internal write-back unified
configuration: level=1

*-cache:1

description: L2 cache
physical id: 6
slot: L2 Cache
size: 1MiB
capacity: 1MiB
capabilities: synchronous internal write-back unified
configuration: level=2

*-cache:2

description: L3 cache
physical id: 7
slot: L3 Cache
size: 6MiB
capacity: 6MiB
capabilities: synchronous internal write-back unified
configuration: level=3

*-memory

description: System Memory
physical id: d
slot: System board or motherboard
size: 8GiB

*-bank:0

description: SODIMM DDR4 Synchronous 2667 MHz (0,4 ns)

```

vendor: 0000
physical id: 0
serial: 00000000
slot: ChannelA-DIMM0
size: 4GiB
width: 64 bits
clock: 2667MHz (0.4ns)
*-bank:1
description: SODIMM DDR4 Synchronous 2667 MHz (0,4 ns)
product: SMS4TDC3C0K0446SCG
vendor: AMD
physical id: 1
serial: 044F7895
slot: ChannelB-DIMM0
size: 4GiB
width: 64 bits
clock: 2667MHz (0.4ns)

```

...

Comando específico para exibir informações da memória RAM de forma mais resumida:

```
$ sudo lshw -C memory -short
```

H/W path	Device	Class	Description
/0/0	memory	128KiB BIOS	
/0/4/5	memory	256KiB L1 cache	
/0/4/6	memory	1MiB L2 cache	
/0/4/7	memory	6MiB L3 cache	
/0/d	memory	8GiB System Memory	
/0/d/0	memory	4GiB SODIMM DDR4 Synchronous	
/0/d/1	memory	4GiB SODIMM DDR4 Synchronous	
/0/100/14.2	memory	RAM memory	

7. Comando lspci

Durante o processo de carga do sistema operacional, o Kernel detecta os dispositivos PCI conectados no computador.

Todos os dispositivos conectados nos barramentos PCI, com todas as suas informações, podem ser listados através do comando "lspci".

```
$ lspci
```

```

00:00.0 Host bridge: Intel Corporation Comet Lake-U v1 4c Host Bridge/DRAM Controller (rev 0c)
00:02.0 VGA compatible controller: Intel Corporation CometLake-U GT2 [UHD Graphics] (rev 02)
00:08.0 System peripheral: Intel Corporation Xeon E3-1200 v5/v6 / E3-1500 v5 / 6th/7th/8th Gen Core Processor Gaussian Mixture Model
00:12.0 Signal processing controller: Intel Corporation Comet Lake Thermal Subsystem
00:14.0 USB controller: Intel Corporation Comet Lake PCH-LP USB 3.1 xHCI Host Controller

```

00:14.2 RAM memory: Intel Corporation Comet Lake PCH-LP Shared SRAM
00:15.0 Serial bus controller: Intel Corporation Serial IO I2C Host Controller
...

As opções mais usadas com o "lspci" são:

- v: Mostra informações detalhadas sobre todos os dispositivos.
- vv: Mostra ainda mais informações sobre os dispositivos.
- n: Mostra os códigos dos fabricantes e dispositivos.
- s: Mostra somente os dispositivos com o slot selecionados.

Comando lspci com filtro para exibir informações da memória:

```
$ lspci | grep memory
```

```
00:14.2 RAM memory: Intel Corporation Comet Lake PCH-LP Shared SRAM
```

Comando lspci com filtro para exibir informações da memória de vídeo:

```
$ lspci | grep VGA
```

```
00:02.0 VGA compatible controller: Intel Corporation CometLake-U GT2 [UHD Graphics]  
(rev 02)
```

```
$ lspci -v -s 00:14.2
```

```
00:14.2 RAM memory: Intel Corporation Comet Lake PCH-LP Shared SRAM  
  Subsystem: Acer Incorporated [ALI] Comet Lake PCH-LP Shared SRAM  
  Flags: bus master, fast devsel, latency 0  
  Memory at b149c000 (64-bit, non-prefetchable) [size=8K]  
  Memory at b14a0000 (64-bit, non-prefetchable) [size=4K]  
  Capabilities: <access denied>
```

```
$ lspci -v -s 00:02.0
```

```
00:02.0 VGA compatible controller: Intel Corporation CometLake-U GT2 [UHD Graphics]  
(rev 02) (prog-if 00 [VGA controller])  
  Subsystem: Acer Incorporated [ALI] CometLake-U GT2 [UHD Graphics]  
  Flags: bus master, fast devsel, latency 0, IRQ 130  
  Memory at b0000000 (64-bit, non-prefetchable) [size=16M]  
  Memory at a0000000 (64-bit, prefetchable) [size=256M]  
  I/O ports at 4000 [size=64]  
  Expansion ROM at 000c0000 [virtual] [disabled] [size=128K]  
  Capabilities: <access denied>  
  Kernel driver in use: i915  
  Kernel modules: i915
```

8. Arquivo `/proc/interrupts`

Usado para consultar a interrupção dos dispositivos.

Exemplo:

```
$ cat /proc/interrupts
```

```
          CPU0      CPU1      CPU2      CPU3      CPU4      CPU5  
          CPU6      CPU7  
1: 172235 0 0 0 8604 0 0 0 IR-IO-APIC  
1-edge i8042
```

```

8:      0      0      0      0      0      0      0      0      0 IR-IO-APIC 8-edge
    rtc0
9:    214      6      0      0      0      0      0      0      0 IR-IO-APIC 9-fasteoi
acpi
14:  3283130      0      0      0      0      0      0      131989      0      IR-IO-APIC
14-fasteoi INT34BB:00
16:      0      0      0      0      0      0      0      0      1293      IR-IO-APIC
16-fasteoi i801_smbus, id

```

Comando específico para exibir a interrupção da memória:

```
$ cat /proc/interrupts | grep dmar
```

```

120: 0      0      0      0      0      0      0      0      0 DMAR-MSI 0-edge dmar0
121: 0      0      0      0      0      0      0      0      0 DMAR-MSI 1-edge dmar1

```

9. Arquivo /proc/ioports

Usado para consultar as portas de entrada e saída (ioports) do hardware.

Exemplo:

```
$ cat /proc/ioports
```

```
$ cat /proc/ioports
```

```
0000-0000 : PCI Bus 0000:00
```

```
0000-0000 : dma1
```

```
0000-0000 : pic1
```

```
0000-0000 : timer0
```

```
0000-0000 : timer1
```

```
0000-0000 : keyboard
```

```
0000-0000 : PNP0C09:00
```

```
0000-0000 : EC data
```

```
0000-0000 : keyboard
```

```
0000-0000 : PNP0C09:00
```

```
0000-0000 : EC cmd
```

```
0000-0000 : rtc0
```

```
0000-0000 : dma page reg
```

```
0000-0000 : pic2
```

```
0000-0000 : dma2
```

```
...
```