

Monitorowanie zasobów: free, vmstat, iostat, sara

Monitorowanie dostępnej pamięci jest kluczowe dla stabilności systemu. Przy administracji serwerami, maszyn wirtualnych czy serwerów ważne jest zrozumienie jak Linux alokuje i zarządza pamięcią.

Komenda `free` - stan pamięci teraz

```
free
```

```
root@rivia:~$ free
              total        used        free      shared  buff/cache   available
Mem:          107171548    21763156    12323136    163080    74416692    85408392
Swap:           8388604      127760     8260844
```

Wyświetla podsumowanie zużycia RAM i swap. Domyślnie w kilobajtach (KB).

```
free -h
```

Format czytelny (B, KB, MB, GB). Najczęściej stosowana opcja.

```
root@rivia:~$ free -h
              total        used        free      shared  buff/cache   available
Mem:          102Gi         20Gi         11Gi         159Mi        70Gi         81Gi
Swap:           8.0Gi        124Mi         7.9Gi
```

```
free -m
```

```
root@rivia:~$ free -m
              total        used        free      shared  buff/cache   available
Mem:          104659        21281        12006         159        72672        83378
Swap:           8191          124          8067
```

Wynik w megabajtach.

```
free -g
```

```
root@rivia:~$: free -g
              total        used        free      shared  buff/cache   available
Mem:           102          20          11           0           70          81
Swap:           7           0           7
root@rivia:~$: █
```

Wynik w gigabajtach (przydatne na dużych serwerach).

```
free -w
```

Rozszerzony format z osobną kolumną na bufor i cache.

```
root@rivia:~$: free -w
              total        used        free      shared  buffers      cache   available
Mem:    107171548    21758316    12328072    163080    773348    73643248    85413232
Swap:     8388604     127760     8260844
root@rivia:~$: █
```

Interpretacja wyniku `free -h`:

```
total used free shared buff/cache available Mem: 15.6G 8.2G 2.1G 0.5G 5.3G 7.4G Swap: 4.0G 0.5G 3.5G
```

```
root@rivia:~$: free -h
              total        used        free      shared  buff/cache   available
Mem:           102Gi         20Gi         11Gi         159Mi         70Gi         81Gi
Swap:           8.0Gi         124Mi         7.9Gi
```

- `total` – całkowita pamięć RAM
- `used` – faktycznie używana pamięć
- `free` – zupełnie wolna (często mała, bo Linux używa cache)
- `buff/cache` – bufor i cache (mogą być zwolnione, jeśli trzeba)
- `available` – pamięć dostępna do użytku dla aplikacji (ważna liczba!)

Monitorowanie pamięci w czasie rzeczywistym

```
free -h -s 2
```

Wyświetla stan pamięci co 2 sekundy.

```

root@rivia:~$ free -h -s 2
              total        used        free      shared  buff/cache   available
Mem:          102Gi        20Gi         11Gi        159Mi        70Gi         81Gi
Swap:         8.0Gi         124Mi         7.9Gi

              total        used        free      shared  buff/cache   available
Mem:          102Gi        20Gi         11Gi        159Mi        70Gi         81Gi
Swap:         8.0Gi         124Mi         7.9Gi

              total        used        free      shared  buff/cache   available
Mem:          102Gi        20Gi         11Gi        159Mi        70Gi         81Gi
Swap:         8.0Gi         124Mi         7.9Gi

```

```
free -h -s 2 -c 10
```

Monitoruje co 2 sekundy, 10 razy.

```

root@rivia:~$ free -h -s 2 -c 3
              total        used        free      shared  buff/cache   available
Mem:          102Gi        20Gi         11Gi        159Mi        70Gi         81Gi
Swap:         8.0Gi         124Mi         7.9Gi

              total        used        free      shared  buff/cache   available
Mem:          102Gi        20Gi         11Gi        159Mi        70Gi         81Gi
Swap:         8.0Gi         124Mi         7.9Gi

              total        used        free      shared  buff/cache   available
Mem:          102Gi        20Gi         11Gi        159Mi        70Gi         81Gi
Swap:         8.0Gi         124Mi         7.9Gi
root@rivia:~$ █

```

```
watch -n 1 'free -h'
```

Monitorowanie pamięci co sekundę w atrakcyjnym formacie (wymaga pakietu `watch`).

```

Every 1.0s: free -h                                rivia: Sun Nov 23 19:41:42 2025
              total        used        free      shared  buff/cache   available
Mem:          102Gi        20Gi         11Gi        159Mi        70Gi         81Gi
Swap:         8.0Gi         124Mi         7.9Gi

```

Komenda `vmstat` – Virtual Memory Statistics

Wprowadzenie do `vmstat`:

Bardziej zaawansowana niż `free`, pokazuje statystyki pamięci wirtualnej, I/O i swap activity.

```
vmstat
```

```
root@rivia:~$: vmstat
procs -----memory----- ---swap-- -----io----- -system-- -----cpu-----
 r b  swpd  free  buff  cache  si  so   bi  bo   in  cs us sy id wa st gu
 1 0 127760 12299732 773348 73643936 0 3 251 627 5993 2 0 0 99 0 0 0
root@rivia:~$: █
```

Jednorazowy raport.

```
vmstat 2 5
```

Raport co 2 sekundy, 5 razy.

```
root@rivia:~$: vmstat 2 3
procs -----memory----- ---swap-- -----io----- -system-- -----cpu-----
 r b  swpd  free  buff  cache  si  so   bi  bo   in  cs us sy id wa st gu
 1 0 127760 12306000 773348 73644016 0 3 251 627 5993 2 0 0 99 0 0 0
 0 0 127760 12309644 773348 73644020 0 0 0 132 5024 7471 0 0 99 0 0 0
 0 0 127760 12315660 773348 73644024 0 0 0 254 5098 8531 0 0 99 0 0 0
root@rivia:~$: █
```

Interpretacja wyniku:

```
procs -----memory----- ---swap-- -----io----- -system-- -----cpu----- r b swpd free buff cache si so bi bo in cs us sy id wa st gu
us sy id wa st 1 0 0 2048000 51200 5242880 0 0 2 1 50 30 25 5 70 0 0
```

```
root@rivia:~$: vmstat
procs -----memory----- ---swap-- -----io----- -system-- -----cpu-----
 r b  swpd  free  buff  cache  si  so   bi  bo   in  cs us sy id wa st gu
 1 0 127760 12299732 773348 73643936 0 3 251 627 5993 2 0 0 99 0 0 0
root@rivia:~$: █
```

- `r` - procesy czekające na CPU
- `b` - procesy w I/O wait
- `swpd` - swap w użyciu (MB)
- `free` - wolna pamięć
- `si/so` - swap in/out (jeśli > 0, system ma problemy z pamięcią)
- `id` - % czasu CPU bezczynny (wysoki = dobry, niski = CPU obciążony)
- `wa` - % czasu czekania na I/O (wysokie wartości = problemy z dyskiem)

Komenda `iostat` - I/O Statistics

`iostat` monitoruje aktywność dysków i procesorów. Wymaga pakietu `sysstat`.

```
iostat
```

Ogólne statystyki od ostatniego restartu systemu.

```
konio@metin2-srv:~$ iostat
          tty          ada0          ada1          ada2          cpu
tin  tout KB/t   tps  MB/s KB/t   tps  MB/s KB/t   tps  MB/s   us ni sy in id
  0    28 24.0    1  0.02 33.1    0  0.00 14.4    0  0.00   0  0  0  0 99
konio@metin2-srv:~$ █
```

```
iostat -x 2 5
```

Rozszerzone statystyki dysku, co 2 sekundy, 5 razy.

```

konio@metin2-srv:~$: iostat -x 2 3
                    extended device statistics
device      r/s      w/s      kr/s      kw/s  ms/r  ms/w  ms/o  ms/t  qlen  %b
ada0         0         1         0.5      21.5    2     3     0     3     0     0
ada1         0         0         0.1       0.3    2     3     0     3     0     0
ada2         0         0         2.1       2.1    2     7     0     3     0     0
ada3         0         0         0.0       0.0    0     0     0     0     0     0
ada4         0         0         0.0       1.0    6     3     0     4     0     0
cd0          0         0         0.0       0.0    0     0     1     1     0     0
pass0        0         0         0.0       0.0    0     0     0     0     0     0
pass1        0         0         0.0       0.0    0     0     0     0     0     0
pass2        0         0         0.0       0.0    0     0     0     0     0     0
pass3        0         0         0.0       0.0    0     0     0     0     0     0
pass4        0         0         0.0       0.0    0     0     0     0     0     0
pass5        0         0         0.0       0.0    0     0     0     0     0     0

                    extended device statistics
device      r/s      w/s      kr/s      kw/s  ms/r  ms/w  ms/o  ms/t  qlen  %b
ada0         0         3         0.0      80.2    0     1     0     1     0     0
ada1         0         0         0.0       0.0    0     0     0     0     0     0
ada2         0         1         0.0      48.0    0    63     0    63     0     3
ada3         0         0         0.0       0.0    0     0     0     0     0     0
ada4         0         0         0.0       0.0    0     0     0     0     0     0
cd0          0         0         0.0       0.0    0     0     0     0     0     0
pass0        0         0         0.0       0.0    0     0     0     0     0     0
pass1        0         0         0.0       0.0    0     0     0     0     0     0
pass2        0         0         0.0       0.0    0     0     0     0     0     0
pass3        0         0         0.0       0.0    0     0     0     0     0     0
pass4        0         0         0.0       0.0    0     0     0     0     0     0
pass5        0         0         0.0       0.0    0     0     0     0     0     0

                    extended device statistics
device      r/s      w/s      kr/s      kw/s  ms/r  ms/w  ms/o  ms/t  qlen  %b
ada0         0         0         0.0       0.0    0     0     0     0     0     0
ada1         0         0         0.0       0.0    0     0     0     0     0     0
ada2         0         0         0.0       0.0    0     0     0     0     0     0
ada3         0         0         0.0       0.0    0     0     0     0     0     0
ada4         0         0         0.0       0.0    0     0     0     0     0     0
cd0          0         0         0.0       0.0    0     0     0     0     0     0
pass0        0         0         0.0       0.0    0     0     0     0     0     0
pass1        0         0         0.0       0.0    0     0     0     0     0     0
pass2        0         0         0.0       0.0    0     0     0     0     0     0
pass3        0         0         0.0       0.0    0     0     0     0     0     0
pass4        0         0         0.0       0.0    0     0     0     0     0     0
pass5        0         0         0.0       0.0    0     0     0     0     0     0
konio@metin2-srv:~$: █

```

Interpretacja wyniku:

Device r/s w/s rMB/s wMB/s rrqm/s wrqm/s %rrqm %wrqm r_await w_await aqu-sz %util sda 10.5 5.2 0.50 0.25
0.5 1.2 5% 10% 2.5ms 5.0ms 0.08 8.5%

```

konio@metin2-srv:~$: iostat
          tty          ada0          ada1          ada2          cpu
tin  tout KB/t    tps  MB/s KB/t    tps  MB/s KB/t    tps  MB/s  us ni  sy in id
   0    28 24.0     1  0.02 33.1     0  0.00 14.4     0  0.00   0  0   0  0 99
konio@metin2-srv:~$: █

```

- `r/s` - read operations per second
- `w/s` - write operations per second
- `rMB/s / wMB/s` - megabajty czytane/zapisywane na sekundę
- `%util` - procent wykorzystania dysku (>80% = problem)
- `r_await / w_await` - średni czas oczekiwania operacji (wysoki = wolny dysk)

Komenda `sar` - System Activity Reporter

`sar` z pakietu `sysstat` to kompleksowe narzędzie do monitorowania systemu. Może zbierać dane historyczne.

```
sar -u 2 5
```

```

root@kon003:~$: sar -u 2 5
Linux 5.14.0-570.30.1.el9_6.x86_64 (kon003.dc.local) 11/23/2025 _x86_64_ (2 CPU)

07:48:14 PM CPU %user %nice %system %iowait %steal %idle
07:48:16 PM all 0.25 0.00 0.25 0.50 0.00 99.00
07:48:18 PM all 0.00 0.00 0.00 0.00 0.00 100.00
07:48:20 PM all 0.25 0.00 0.25 0.00 0.00 99.50
07:48:22 PM all 0.00 0.00 0.00 0.00 0.00 100.00
07:48:24 PM all 0.00 0.00 0.25 0.00 0.00 99.75
Average: all 0.10 0.00 0.15 0.10 0.00 99.65
root@kon003:~$: █

```

Statystyki CPU co 2 sekundy, 5 razy.

```
sar -r 2 5
```

```

root@kon003:~$: sar -r 1 1
Linux 5.14.0-570.30.1.el9_6.x86_64 (kon003.dc.local) 11/23/2025 _x86_64_ (2 CPU)

07:48:49 PM kbmemfree kbavail kbmempused %memused kbbuffers kbcached kbcommit %commit kbactive kbinact kbdirty
07:48:50 PM 1350408 2473544 898328 24.00 217648 1109156 3838068 48.38 1382616 715568 128
Average: 1350408 2473544 898328 24.00 217648 1109156 3838068 48.38 1382616 715568 128

```

Statystyki pamięci.

```
sar -d 2 5
```

```

root@kon003:~$: sar -d 1 1
Linux 5.14.0-570.30.1.el9_6.x86_64 (kon003.dc.local) 11/23/2025 _x86_64_ (2 CPU)

07:48:57 PM      DEV      tps      rkB/s      wkB/s      dkB/s      areq-sz      aqu-sz      await      %util
07:48:58 PM      sda      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
07:48:58 PM      dm-0     0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
07:48:58 PM      dm-1     0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
07:48:58 PM      dm-2     0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00

Average:         DEV      tps      rkB/s      wkB/s      dkB/s      areq-sz      aqu-sz      await      %util
Average:         sda      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
Average:         dm-0     0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
Average:         dm-1     0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
Average:         dm-2     0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00

```

Statystyki dysków (podobnie jak `iotstat`).

```

root@kon003:~$: sar -b 1 1
Linux 5.14.0-570.30.1.el9_6.x86_64 (kon003.dc.local) 11/23/2025 _x86_64_ (2 CPU)

07:49:05 PM      tps      rtps      wtps      dtps      bread/s      bwrtn/s      bdscd/s
07:49:06 PM      0.00      0.00      0.00      0.00      0.00      0.00      0.00
Average:         0.00      0.00      0.00      0.00      0.00      0.00      0.00
root@kon003:~$: █

```

```
sar -b 2 5
```

I/O i przełączenia stron (page faults).

```

root@kon003:~$: sar -b 1 1
Linux 5.14.0-570.30.1.el9_6.x86_64 (kon003.dc.local) 11/23/2025 _x86_64_ (2 CPU)

07:49:05 PM      tps      rtps      wtps      dtps      bread/s      bwrtn/s      bdscd/s
07:49:06 PM      0.00      0.00      0.00      0.00      0.00      0.00      0.00
Average:         0.00      0.00      0.00      0.00      0.00      0.00      0.00
root@kon003:~$: █

```

Monitorowanie swap activity - problem w systemach

```
vmstat 1 | grep -v procs | awk '$7 > 0 || $8 > 0 { print "SWAP AKTYWNY!", $0 }'
```

```

root@kon003:~$: vmstat 1 | grep -v procs | awk '$7 > 0 || $8 > 0 { print "SWAP AKTYWNY!", $0 }'
SWAP AKTYWNY!  r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa  st
SWAP AKTYWNY!  r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa  st
SWAP AKTYWNY!  r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa  st
SWAP AKTYWNY!  r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa  st
SWAP AKTYWNY!  r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa  st

```

Ostrzeżenie gdy system zaczyna pisać do swap

```
watch -n 1 'vmstat 1 2 | tail -1'
```

Ciągle monitorowanie najnowszych danych vmstat.

```
Every 1.0s: vmstat 1 2 | tail -1
```

```
0 0 0 1357688 217688 1218692 0 0 0 0 188 273 0 0 100 0 0
```

Praktyczne problemy z pamięcią i jak je diagnozować

Problem: System powolny, dużo swap activity

1. Sprawdź dostępną pamięć `free -h`

2. Sprawdź swap

```
vmstat 1 5 | grep -E 'si|so'
```

3. Szukaj procesów zużywających pamięć

```
ps aux --sort=-%mem | head -20
```

4. Sprawdź I/O wait

```
iostat -x 1 5
```

Problem: Serwer Metin2 (MT2SRV) zajmuje za dużo pamięci

```
# Monitoruj samo server gry watch -n 1 'ps aux | grep -E "game|db|auth" | awk "{print \$2, \$4, \$11}"'
```

Jeśli pamięć rośnie bezustannie (memory leak)

```
ps aux | grep game | awk '{print $2}' # Uzyskaj PID
```

```
cat /proc/[PID]/status | grep VmRSS # Aktualna pamięć tego procesu
```

Wyzwalanie alertów przy braku pamięci

```
#!/bin/bash AVAILABLE=$(free -h | awk '/^Mem:/ {print $7}' | tr -d 'G') if (( $(echo "$AVAILABLE < 1" | bc -l) ));  
then echo "ALARM: Pamięć poniżej 1GB!" fi
```

Revision #2

Created 23 November 2025 18:36:56 by Tomasz Konieczny

Updated 23 November 2025 18:53:25 by Tomasz Konieczny