

Obsługa dysków i systemów plików: df, du, lsblk, lscsi

Zarządzanie dyskami i miejscem na nich to kluczowa część administracji systemów. W serwerach (szczególnie Proxmox, FreeBSD czy serwery gier) ważne jest monitorowanie pojemności dysków i szybkie diagnozowanie problemów z miejscem na dysku. Gdy dysk się zapełni, serwis przestaje działać – stąd konieczność bieżącego monitorowania i konserwacji.

Komenda `df` – informacje o systemach plików

`df`

Wyświetla użycie dysku dla wszystkich zamontowanych systemów plików (domyślnie w blokach 1K). Jest to podstawowe narzędzie do szybkiego sprawdzenia "ile miejsca zostało".

```
root@kon003:~# df
Filesystem            1K-blocks    Used Available Use% Mounted on
devtmpfs              4096         0      4096    0% /dev
tmpfs                 1871696      0    1871696    0% /dev/shm
tmpfs                 748680     21740    726940    3% /run
/dev/mapper/vgROOT-lvROOT 12278920 2672512    8960880   23% /
/dev/sda1             983040     335316    647724   35% /boot
/dev/mapper/vgROOT-lvDOCKER 15371208 9655820    4912780   67% /docker
overlay               15371208 9655820    4912780   67% /docker/docker-data/overlay2/68fc332cb012f04c60c813b70a35da0efe8d7a76563a23456a44d1322f3a8a27/merged
overlay               15371208 9655820    4912780   67% /docker/docker-data/overlay2/6a99598cea0a290d8c84a0c45adc9de151c3a93daf83d5ded5f75f46ad187155/merged
overlay               15371208 9655820    4912780   67% /docker/docker-data/overlay2/5ad56559ad9d0c17e70d5f6a3cb5a8b8fd44eb604d6b2bb2ff4c804bcb17aa/merged
overlay               15371208 9655820    4912780   67% /docker/docker-data/overlay2/5ccee3ca8673c7cc666ebc37ce18bc23b802d04a5f2b1acdda778dd469631d9f/merged
overlay               15371208 9655820    4912780   67% /docker/docker-data/overlay2/f141275f39bca281c37d840559d2792468e5d336435cca2c32e8821d9e41balf/merged
overlay               15371208 9655820    4912780   67% /docker/docker-data/overlay2/ce10e8588bc791c2bd42310b5842434506369f8e93252e54c2af8ca1c98ef035/merged
tmpfs                 374336         0     374336    0% /run/user/0
overlay               15371208 9655820    4912780   67% /docker/docker-data/overlay2/4fe72de911813ad84b648f7d3c0b82db9cd6f6876818fcb5adac73ea89eb14d6/merged
```

`df -h`

Format czytelny (B, KB, MB, GB) – najczęstszy sposób użycia. Zamiast liczb takich jak 15360000 zobaczymy 15G.

```

root@kon003:~$: df -h
Filesystem                Size      Used Avail Use% Mounted on
devtmpfs                  4.0M         0  4.0M   0% /dev
tmpfs                      1.8G         0  1.8G   0% /dev/shm
tmpfs                      732M       22M   710M   3% /run
/dev/mapper/vgROOT-lvROOT  12G       2.6G   8.6G  23% /
/dev/sda1                  960M       328M   633M  35% /boot
/dev/mapper/vgROOT-lvDOCKER 15G       9.3G   4.7G  67% /docker
overlay                    15G       9.3G   4.7G  67% /docker/docker-data/overlay2/68fc332cb012
overlay                    15G       9.3G   4.7G  67% /docker/docker-data/overlay2/6a99598cea0a
overlay                    15G       9.3G   4.7G  67% /docker/docker-data/overlay2/5ad56559ad9d
overlay                    15G       9.3G   4.7G  67% /docker/docker-data/overlay2/5ccee3ca8673
overlay                    15G       9.3G   4.7G  67% /docker/docker-data/overlay2/f141275f39bc
overlay                    15G       9.3G   4.7G  67% /docker/docker-data/overlay2/ce10e8588bc7
tmpfs                      366M         0  366M   0% /run/user/0
overlay                    15G       9.3G   4.7G  67% /docker/docker-data/overlay2/4fe72de91181

```

```
df -H
```

Format dziesiętny (1000 zamiast 1024 jako podstawa) - dla jednostek SI. Niektórzy wolą taki zapis.

```

root@kon003:~$: df -H
Filesystem                Size      Used Avail Use% Mounted on
devtmpfs                  4.2M         0  4.2M   0% /dev
tmpfs                      2.0G         0  2.0G   0% /dev/shm
tmpfs                      767M       23M   745M   3% /run
/dev/mapper/vgROOT-lvROOT  13G       2.8G   9.2G  23% /
/dev/sda1                  1.1G       344M   664M  35% /boot
/dev/mapper/vgROOT-lvDOCKER 16G       9.9G   5.1G  67% /docker
overlay                    16G       9.9G   5.1G  67% /docker/docker-data/d
overlay                    16G       9.9G   5.1G  67% /docker/docker-data/d
overlay                    16G       9.9G   5.1G  67% /docker/docker-data/d
overlay                    16G       9.9G   5.1G  67% /docker/docker-data/d
overlay                    16G       9.9G   5.1G  67% /docker/docker-data/d
tmpfs                      384M         0  384M   0% /run/user/0
overlay                    16G       9.9G   5.1G  67% /docker/docker-data/d

```

```
df -T
```

Dodaje kolumnę z typem systemu plików (ext4, xfs, btrfs, zfs, ntfs, etc.). Przydatne gdy zarządzasz różnymi typami systemów plików.

```

root@kon003:~$: df -Th
Filesystem                Type      Size  Used Avail Use% Mounted on
devtmpfs                  devtmpfs  4.0M   0    4.0M  0%  /dev
tmpfs                     tmpfs     1.8G   0    1.8G  0%  /dev/shm
tmpfs                     tmpfs     732M  22M   710M  3%  /run
/dev/mapper/vgROOT-lvROOT ext4       12G   2.6G   8.6G  23%  /
/dev/sda1                 xfs        960M  328M   633M  35%  /boot
/dev/mapper/vgROOT-lvDOCKER ext4       15G   9.3G   4.7G  67%  /docker
overlay                   overlay    15G   9.3G   4.7G  67%  /docker/docker-data/o
overlay                   overlay    15G   9.3G   4.7G  67%  /docker/docker-data/o
overlay                   overlay    15G   9.3G   4.7G  67%  /docker/docker-data/o
overlay                   overlay    15G   9.3G   4.7G  67%  /docker/docker-data/o
overlay                   overlay    15G   9.3G   4.7G  67%  /docker/docker-data/o
overlay                   overlay    15G   9.3G   4.7G  67%  /docker/docker-data/o
tmpfs                     tmpfs     366M   0    366M  0%  /run/user/0
overlay                   overlay    15G   9.3G   4.7G  67%  /docker/docker-data/o

```

```
df -i
```

Wyświetla inode'y zamiast bloków. Ważne: można mieć wolne miejsce, ale brak inode'ów (gdy mamy miliony małych plików). System wtedy nie pozwoli tworzyć nowych plików mimo wolnego miejsca!

```

root@kon003:~$: df -ih
Filesystem                Inodes  IUsed  IFree  IUse% Mounted on
devtmpfs                  452K    388    452K   1%  /dev
tmpfs                     457K     1    457K   1%  /dev/shm
tmpfs                     800K    846    800K   1%  /run
/dev/mapper/vgROOT-lvROOT 768K    60K    709K   8%  /
/dev/sda1                 512K    366    512K   1%  /boot
/dev/mapper/vgROOT-lvDOCKER 960K   349K    612K  37%  /docker
overlay                   960K   349K    612K  37%  /docker/docker-d
overlay                   960K   349K    612K  37%  /docker/docker-d
overlay                   960K   349K    612K  37%  /docker/docker-d
overlay                   960K   349K    612K  37%  /docker/docker-d
overlay                   960K   349K    612K  37%  /docker/docker-d
tmpfs                     92K     15     92K   1%  /run/user/0
overlay                   960K   349K    612K  37%  /docker/docker-d
root@kon003:~$: █

```

Praktyczne kombinacje:

```
df -Th
```

Typ systemu plików i format czytelny – najbardziej praktyczna forma dla administratorów.

```

root@kon003:~$ df -Th
Filesystem                Type      Size  Used Avail Use% Mounted on
devtmpfs                  devtmpfs  4.0M   0    4.0M   0% /dev
tmpfs                     tmpfs     1.8G   0    1.8G   0% /dev/shm
tmpfs                     tmpfs     732M   22M   710M   3% /run
/dev/mapper/vgROOT-lvROOT ext4       12G   2.6G   8.6G  23% /
/dev/sda1                 xfs        960M   328M   633M  35% /boot
/dev/mapper/vgROOT-lvDOCKER ext4       15G   9.3G   4.7G  67% /docker
overlay                   overlay    15G   9.3G   4.7G  67% /docker/docker-data/overlay2/68fc332cb0
overlay                   overlay    15G   9.3G   4.7G  67% /docker/docker-data/overlay2/6a99598cea
overlay                   overlay    15G   9.3G   4.7G  67% /docker/docker-data/overlay2/5ad56559ad
overlay                   overlay    15G   9.3G   4.7G  67% /docker/docker-data/overlay2/5ccee3ca86
overlay                   overlay    15G   9.3G   4.7G  67% /docker/docker-data/overlay2/f141275f39
overlay                   overlay    15G   9.3G   4.7G  67% /docker/docker-data/overlay2/ce10e8588b
tmpfs                     tmpfs     366M   0    366M   0% /run/user/0
overlay                   overlay    15G   9.3G   4.7G  67% /docker/docker-data/overlay2/4fe72de911

```

```
df -h | grep -v tmpfs
```

Ukryj tymczasowe systemy plików tmpfs, devtmpfs (często nieistotne).

```

root@kon003:~$ df -h | grep tmpfs
devtmpfs                4.0M   0    4.0M   0% /dev
tmpfs                   1.8G   0    1.8G   0% /dev/shm
tmpfs                   732M   22M   710M   3% /run
tmpfs                   366M   0    366M   0% /run/user/0
root@kon003:~$ █

```

```
df -h | awk '$5 > 80 {print $0}'
```

Pokaż tylko dyski wypełnione powyżej 80% (kolumna 5 to procent użycia).

```

root@kon003:~$ df -h | awk '$5 > 30 {print $0}'
Filesystem                Size  Used Avail Use% Mounted on
/dev/sda1                 960M   328M   633M  35% /boot
/dev/mapper/vgROOT-lvDOCKER 15G   9.3G   4.7G  67% /docker
overlay                   15G   9.3G   4.7G  67% /docker/docker-data/overlay2/68fc332cb0
overlay                   15G   9.3G   4.7G  67% /docker/docker-data/overlay2/6a99598cea
overlay                   15G   9.3G   4.7G  67% /docker/docker-data/overlay2/5ad56559ad
overlay                   15G   9.3G   4.7G  67% /docker/docker-data/overlay2/5ccee3ca86
overlay                   15G   9.3G   4.7G  67% /docker/docker-data/overlay2/f141275f39
overlay                   15G   9.3G   4.7G  67% /docker/docker-data/overlay2/ce10e8588b
overlay                   15G   9.3G   4.7G  67% /docker/docker-data/overlay2/4fe72de911
root@kon003:~$ █

```

```
df -hT | sort -k6 -rn
```

Posortuj dyski wg procentu zajęcia miejsca (kolumna 6) malejąco - szybko znajdziesz problematyczne dyski.

```

root@kon003:~$: df -hT | sort -k6 -rn
overlay                overlay      15G   9.3G   4.7G   67% /docker/docker-data/ov
overlay                overlay      15G   9.3G   4.7G   67% /docker/docker-data/ov
overlay                overlay      15G   9.3G   4.7G   67% /docker/docker-data/ov
overlay                overlay      15G   9.3G   4.7G   67% /docker/docker-data/ov
overlay                overlay      15G   9.3G   4.7G   67% /docker/docker-data/ov
overlay                overlay      15G   9.3G   4.7G   67% /docker/docker-data/ov
overlay                overlay      15G   9.3G   4.7G   67% /docker/docker-data/ov
/dev/mapper/vgROOT-lvDOCKER ext4         15G   9.3G   4.7G   67% /docker
/dev/sda1              xfs          960M   328M   633M   35% /boot
/dev/mapper/vgROOT-lvROOT ext4         12G    2.6G   8.6G   23% /
tmpfs                  tmpfs        732M    22M   710M    3% /run
tmpfs                  tmpfs        366M     0    366M    0% /run/user/0
tmpfs                  tmpfs        1.8G     0    1.8G    0% /dev/shm
Filesystem             Type        Size  Used Avail Use% Mounted on
devtmpfs               devtmpfs    4.0M     0   4.0M    0% /dev

```

Komenda `du` – rozmiar katalogów i plików

`du` (disk usage) pokazuje ile miejsca zajmują poszczególne katalogi i pliki. Jest niezbędne gdy `df` pokazuje że dysk jest pełny, a potrzebujemy znaleźć które katalogi lub pliki odpowiadają za ten stan.

```
du
```

```

root@kon003:/var/log$: du
4      ./sa
18988 ./audit
4      ./private
4      ./chrony
7028   ./anaconda
4      ./qemu-ga
1472   ./rhm
4      ./puppetlabs/pxp-agent
4      ./puppetlabs/puppet
12     ./puppetlabs
4      ./samba/old
8      ./samba
4424   ./sssd
51904  .
root@kon003:/var/log$: █

```

Domyślnie wyświetla rozmiary wszystkich podkatalogów rekurencyjnie (może wypisać tysiące linii!).

```
du -h
```

Format czytelny (KB, MB, GB).

```
root@kon003:/var$: du -h | head -n30
16K      ./spool/anacron
4.0K    ./spool/rhsm/debug
8.0K    ./spool/rhsm
4.0K    ./spool/lpd
4.0K    ./spool/mail
4.0K    ./spool/cron
40K     ./spool
4.0K    ./nis
4.0K    ./adm
4.0K    ./kerberos/krb5/user
8.0K    ./kerberos/krb5
12K     ./kerberos
4.0K    ./db/sudo/lectured
8.0K    ./db/sudo
12K     ./db
4.0K    ./local
4.0K    ./yp
4.0K    ./log/sa
19M     ./log/audit
4.0K    ./log/private
4.0K    ./log/chrony
6.9M    ./log/anaconda
4.0K    ./log/gemu-ga
1.5M    ./log/rhsm
4.0K    ./log/puppetlabs/pxp-agent
4.0K    ./log/puppetlabs/puppet
12K     ./log/puppetlabs
4.0K    ./log/samba/old
8.0K    ./log/samba
4.4M    ./log/sss
root@kon003:/var$: █
```

```
du -sh katalog/
```

Podsumowanie (-s, summary) rozmiaru katalogu w czytelnym formacie.

```
root@kon003:/var$: du -sh /var/log
51M     /var/log
root@kon003:/var$: █
```

Wszystkie pliki i katalogi (-a, all) z rozmiarami – pokaże też pojedyncze pliki, nie tylko katalogi.

```
du -d 1 -h /var
```

Ograniczenie głębokości (-d 1) do jednego poziomu - nie wchodzi głębiej w strukturę.

```
root@kon003:/var$: du -d 1 -h /var
40K    /var/spool
4.0K   /var/nis
4.0K   /var/adm
12K    /var/kerberos
12K    /var/db
4.0K   /var/local
4.0K   /var/yp
51M    /var/log
44K    /var/tmp
4.0K   /var/ftp
4.0K   /var/preserve
102M   /var/cache
4.0K   /var/opt
4.0K   /var/crash
4.0K   /var/games
4.0K   /var/empty
159M   /var/lib
311M   /var
root@kon003:/var$:
```

Znalezienie największych katalogów i plików:

```
du -ah /var | sort -rh | head -20
```

20 największych plików i podkatalogów w /var (sortuj malejąco wg rozmiaru czytelny dla człowieka).

```
du -sh /* 2>/dev/null | sort -rh
```

Sprawdź rozmiar wszystkich głównych katalogów systemu (/ , /home, /var, /usr itd), ignoruj błędy dostępu.

```
find /var -type f -size +100M -exec ls -lh {} \; | awk '{print $5, $9}' | sort -rh
```

```
root@kon003:/var$: find /var -type f -size +100M -exec ls -lh {} \; | awk '{print $5, $9}' | sort -rh
466G /var/log/lastlog
134M /var/lib/rpm/rpmdb.sqlite
root@kon003:/var$:
```

Znajdź wszystkie pliki większe niż 100MB. Przydatne gdy szukasz co zabiera miejsce.

```
find /var/log -type f -size +1G
```

Znajdź pliki logów większe niż 1GB - często to one zapełniają dysk.

```
root@kon003:/var$: find /var/log -type f -size +1G
/var/log/lastlog
root@kon003:/var$: █
```

Praktyczne scenariusze z doświadczeń sysadmina:

```
# Dysk pełny na serwerze - diagnoza krok po kroku
df -h          # Sprawdź który dysk jest pełny
du -sh /* | sort -rh    # Szukaj dużych katalogów głównych
du -sh /var/* | sort -rh # Zagłębienie w /var
du -sh /var/log/* | sort -rh # Sprawdź logi
```

Sprawdź rozmiar bazy danych

```
du -sh /var/lib/mysql
du -sh /var/lib/postgresql
```

Sprawdź cache, tymczasowe pliki

```
du -sh /tmp /var/tmp /var/cache
```

Komenda `lsblk` - lista urządzeń blokowych

Wprowadzenie:

`lsblk` wyświetla strukturę dysków twardych, partycji, wolumenów logicznych (LVM) oraz punktów montowania. Niezastąpione gdy chcesz szybko zobaczyć "co jest podpięte i gdzie".

```
lsblk
```

Wyświetla wszystkie urządzenia blokowe w formie drzewa. Idealnie pokazuje relacje między dyskami, partycjami i punktami montowania.

```

root@kon003:/var$: lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda                                  8:0      0   32G  0 disk
├─sda1                               8:1      0    1G  0 part /boot
└─sda2                               8:2      0   31G  0 part
   ├─vgROOT-lvROOT                   253:0    0   12G  0 lvm  /
   ├─vgROOT-lvSWAP                   253:1    0    4G  0 lvm  [SWAP]
   └─vgROOT-lvDOCKER                 253:2    0   15G  0 lvm  /docker
sr0                                  11:0     1 1024M  0 rom
root@kon003:/var$: █

```

```
lsblk -f
```

Dodatkowo pokazuje systemy plików, UUID, etykiety (labels) i procent użycia.

```

root@kon003:/var$: lsblk -f
NAME                                FSTYPE      FSVER    LABEL UUID                                FSAVAIL FSUSE% MOUNTPOINTS
sda
├─sda1                             xfs
└─sda2                             LVM2_member LVM2 001  ZP1Lon-LGF5-1003-Lcgf-ZsAv-oU07-de2fDD 632.5M  34% /boot
   ├─vgROOT-lvROOT                  ext4        1.0      3134c3aa-d347-4820-9edc-2b256e1499c6 8.5G    22% /
   ├─vgROOT-lvSWAP                  swap        1        4b5d8ff0-ae13-4a62-8463-70a3be686c14 4.7G    63% [SWAP]
   └─vgROOT-lvDOCKER                ext4        1.0      5cfe9b80-3203-46f8-a5b1-cf1191b98b93 4.7G    63% /docker
sr0
root@kon003:/var$: █

```

```
lsblk -o NAME,SIZE,TYPE,MOUNTPOINT,FSTYPE
```

Wybrane kolumny - dostosuj wyświetlanie do swoich potrzeb.

```

root@kon003:/var$: lsblk -o NAME,SIZE,TYPE,MOUNTPOINT,FSTYPE
NAME                                SIZE TYPE MOUNTPOINT FSTYPE
sda                                  32G disk
├─sda1                               1G part /boot      xfs
└─sda2                               31G part                LVM2_member
   ├─vgROOT-lvROOT                   12G lvm  /          ext4
   ├─vgROOT-lvSWAP                   4G lvm  [SWAP]     swap
   └─vgROOT-lvDOCKER                 15G lvm  /docker    ext4
sr0                                  1024M rom
root@kon003:/var$: █

```

```
lsblk -p
```

Pokazuje pełne ścieżki urządzeń (np. /dev/sda zamiast sda).

```

root@kon003:/var$: lsblk -p
NAME                                MAJ:MIN RM  SIZE RO  TYPE MOUNTPOINTS
/dev/sda                             8:0      0   32G  0  disk
├─/dev/sda1                          8:1      0    1G  0  part /boot
└─/dev/sda2                          8:2      0   31G  0  part
   ├─/dev/mapper/vgROOT-lvROOT       253:0    0   12G  0  lvm  /
   ├─/dev/mapper/vgROOT-lvSWAP       253:1    0    4G  0  lvm  [SWAP]
   └─/dev/mapper/vgROOT-lvDOCKER     253:2    0   15G  0  lvm  /docker
/dev/sr0                             11:0     1 1024M  0  rom

```

Komenda `lsblk` – lista urządzeń SCSI/SATA

```
lsblk
```

Wyświetla listę podłączonych urządzeń SCSI/SATA. Przydatne do weryfikacji że wszystkie dyski fizyczne są widoczne w systemie.

```

root@kon003:/var$: lsscsi
[0:0:0:0]    disk      QEMU      QEMU HARDDISK    2.5+  /dev/sda
[2:0:0:0]    cd/dvd   QEMU      QEMU DVD-ROM     2.5+  /dev/sr0
root@kon003:/var$: █

```

```
lsscsi -s
```

Dodaje rozmiar dysków.

```

root@kon003:/var$: lsscsi -s
[0:0:0:0]    disk      QEMU      QEMU HARDDISK    2.5+  /dev/sda    34.3GB
[2:0:0:0]    cd/dvd   QEMU      QEMU DVD-ROM     2.5+  /dev/sr0    1.07GB
root@kon003:/var$: █

```

Monitorowanie I/O dysku w czasie rzeczywistym

```
iotop -xz 1
```

Monitoruj operacje I/O na dyskach co sekundę (wymaga pakietu `sysstat`). Szukaj kolumny `%util > 80%` – oznacza mocne obciążenie dysku.

```

root@kon003:/var# iostat -xz 1
Linux 5.14.0-570.30.1.el9_6.x86_64 (kon003.dc.local) 11/23/2025 _x86_64_ (2 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.32    0.01    0.24    0.05    0.00   99.38

Device:            r/s    kB/s    rrqm/s    %rrqm  r_await  rareq-sz    w/s    kB/s    wrqm/s    %wrqm  w_await  wareq-sz    d/s    kB/s    drqm/s    %drqm  d_await  dareq-sz    f/s  f_await  aqu-sz  %util
dm-0                0.02    0.36    0.00    0.00    0.50    20.90    0.65    3.59    0.00    0.00    2.22    5.48    0.00    0.00    0.00    0.00    0.00    0.00    0.00  0.00    0.00    0.13
dm-1                0.00    0.00    0.00    0.00    0.26    22.65    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00  0.00    0.00    0.00
dm-2                0.01    0.17    0.00    0.00    0.93    16.02    0.40    1.89    0.00    0.00    2.53    4.73    0.00    0.00    0.00    0.00    0.00    0.00    0.00  0.00    0.00    1.44
sda                 0.02    0.56    0.01    0.01    0.57    26.01    0.66    5.48    0.40    37.80    2.85    8.35    0.00    0.00    0.00    0.00    0.00    0.00    0.00  0.27    2.49    0.00    0.08

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.50    0.00    1.00    0.00    0.00   98.51

Device:            r/s    kB/s    rrqm/s    %rrqm  r_await  rareq-sz    w/s    kB/s    wrqm/s    %wrqm  w_await  wareq-sz    d/s    kB/s    drqm/s    %drqm  d_await  dareq-sz    f/s  f_await  aqu-sz  %util

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.00    0.00    0.50    0.00    0.00   99.50

Device:            r/s    kB/s    rrqm/s    %rrqm  r_await  rareq-sz    w/s    kB/s    wrqm/s    %wrqm  w_await  wareq-sz    d/s    kB/s    drqm/s    %drqm  d_await  dareq-sz    f/s  f_await  aqu-sz  %util

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.00    0.00    0.00    0.00    0.00  100.00

Device:            r/s    kB/s    rrqm/s    %rrqm  r_await  rareq-sz    w/s    kB/s    wrqm/s    %wrqm  w_await  wareq-sz    d/s    kB/s    drqm/s    %drqm  d_await  dareq-sz    f/s  f_await  aqu-sz  %util

```

iotop

Interaktywne narzędzie pokazujące które procesy generują najwięcej operacji I/O (wymaga instalacji i uprawnień root).

```

Total DISK READ :      0.00 B/s | Total DISK WRITE :      260.72 K/s
Actual DISK READ:      0.00 B/s | Actual DISK WRITE:      521.44 K/s
  TID  PRIO  USER      DISK READ  DISK WRITE  COMMAND
-----
375398 be/4  root       0.00 B/s   260.72 K/s portainer
  1 be/4  root       0.00 B/s    0.00 B/s systemd --switched-root --system --deserialize 31
  2 be/4  root       0.00 B/s    0.00 B/s [kthreadd]
  3 be/4  root       0.00 B/s    0.00 B/s [pool_workqueue_]
  4 be/0  root       0.00 B/s    0.00 B/s [kworker/R-rcu_g]
  5 be/0  root       0.00 B/s    0.00 B/s [kworker/R-sync_]
  6 be/0  root       0.00 B/s    0.00 B/s [kworker/R-slab]
  7 be/0  root       0.00 B/s    0.00 B/s [kworker/R-netns]
  9 be/0  root       0.00 B/s    0.00 B/s [kworker/0:0H-events_highpri]
 11 be/0  root       0.00 B/s    0.00 B/s [kworker/R-mm_pe]
 13 be/4  root       0.00 B/s    0.00 B/s [rcu_tasks_kthre]
 14 be/4  root       0.00 B/s    0.00 B/s [rcu_tasks_rude_]
 15 be/4  root       0.00 B/s    0.00 B/s [rcu_tasks_trace]
 16 be/4  root       0.00 B/s    0.00 B/s [ksoftirqd/0]
 17 be/4  root       0.00 B/s    0.00 B/s [rcu_preempt]
 18 be/4  root       0.00 B/s    0.00 B/s [rcu_exp_par_gp_]
 19 be/4  root       0.00 B/s    0.00 B/s [rcu_exp_gp_kthr]
 20 rt/4  root       0.00 B/s    0.00 B/s [migration/0]
 21 rt/4  root       0.00 B/s    0.00 B/s [idle_inject/0]
 23 be/4  root       0.00 B/s    0.00 B/s [cpuhp/0]
 24 be/4  root       0.00 B/s    0.00 B/s [cpuhp/1]
 25 rt/4  root       0.00 B/s    0.00 B/s [idle_inject/1]
 26 rt/4  root       0.00 B/s    0.00 B/s [migration/1]

```

watch -n 1 'df -h | grep -v tmpfs'

Monitoruj zmiany w zajęciu dysku co sekundę (przydatne gdy coś szybko zabiera miejsce).

Diagnostyka problemów z dyskiem – praktyczne scenariusze

Problem: Dysk nagle zapełniony

```

# Krok 1: Sprawdź który system plików
df -h

# Krok 2: Szukaj dużych katalogów (zaczniij od głównych)
du -sh /* 2>/dev/null | sort -rh | head -10

# Krok 3: Zagłębienie (np. jeśli /var jest duży)

```

```
du -sh /var/* | sort -rh | head -10
```

```
# Krok 4: Sprawdź logi
```

```
du -sh /var/log/*
```

```
find /var/log -type f -size +500M -exec ls -lh {} \;
```

```
# Krok 5: Sprawdź czy nie ma usuniętych plików nadal otwartych
```

```
ls -l | grep deleted
```

Problem: Brak miejsca mimo że pliki nie zajmują tyle

```
# Sprawdź inode'y (mogą się skończyć przy wielu małych plikach)
```

```
df -i
```

```
# Jeśli inode'y na wyczerpaniu, znajdź katalogi z wieloma plikami
```

```
find / -xdev -printf '%h\n' | sort | uniq -c | sort -rn | head -20
```

Problem: Serwer gry MT2SRV - logi zapełniają dysk

```
# Sprawdź rozmiar logów gry
```

```
du -sh /path/to/game/log/*
```

```
# Usuń stare logi (ostrożnie!)
```

```
find /path/to/game/log -name "*.log" -mtime +7 -delete
```

```
# Skompresuj duże logi zamiast usuwać
```

```
find /path/to/game/log -name "*.log" -size +100M -exec gzip {} \;
```

Czyszczenie dysku - bezpieczne metody:

```
# Wyczyść cache pakietów (Debian/Ubuntu)
```

```
apt clean
```

```
apt autoclean
```

```
# Wyczyść cache pakietów (RedHat/CentOS)
```

```
yum clean all
```

```
# Usuń stare kerneli (Ubuntu - ostrożnie!)
```

```
apt autoremove --purge
```

```
# Wyczyść dzienniki systemd starsze niż 7 dni
```

```
journalctl --vacuum-time=7d
```

```
# Wyczyść pliki tymczasowe starsze niż 10 dni
```

```
find /tmp -type f -mtime +10 -delete
```

Monitorowanie trendu zajęcia dysku

```
# Zapisz snapshot dziennie
```

```
df -h > /var/log/disk_usage_$(date +%Y%m%d).txt
```

```
# Porównaj z wczoraj
```

```
diff /var/log/disk_usage_20251122.txt /var/log/disk_usage_20251123.txt
```

```
# Prosty skrypt alertu gdy dysk > 90%
```

```
#!/bin/bash
```

```
USAGE=$(df -h | awk 'NR==2 {print $5}' | tr -d '%')
```

```
if [ $USAGE -gt 90 ]; then
```

```
    echo "ALERT: Dysk zapełniony w $USAGE%" | mail -s "Disk Alert" admin@example.com
```

```
fi
```

Sprawdzanie stanu zdrowia dysku (S.M.A.R.T.) (wymaga pakietu `smartmontools`).

```
smartctl -a /dev/sda
```

```
root@rivia:~$: smartctl -a /dev/sda
smartctl 7.4 2024-10-15 r5620 [x86_64-linux-6.14.11-4-pve] (local build)
Copyright (C) 2002-23, Bruce Allen, Christian Franke, www.smartmontools.org

=== START OF INFORMATION SECTION ===
Device Model:          SATA SSD
Serial Number:         A7F0079A17ED00343287
LU WWN Device Id:     0 000000 0000000000
Firmware Version:     SBFMBB.3
User Capacity:        240,057,409,536 bytes [240 GB]
Sector Size:          512 bytes logical/physical
Rotation Rate:        Solid State Device
Form Factor:          2.5 inches
TRIM Command:         Available
Device is:             Not in smartctl database 7.3/5528
ATA Version is:       ACS-4 (minor revision not indicated)
SATA Version is:      SATA 3.2, 6.0 Gb/s (current: 6.0 Gb/s)
Local Time is:        Sun Nov 23 20:59:20 2025 CET
SMART support is:     Available - device has SMART capability.
SMART support is:     Enabled
```

```
=== START OF READ SMART DATA SECTION ===
SMART Status not supported: Incomplete response, ATA output registers missing
SMART overall-health self-assessment test result: PASSED
Warning: This result is based on an Attribute check.
```

```
General SMART Values:
Offline data collection status:  (0x00) Offline data collection activity
                                   was never started.
                                   Auto Offline Data Collection: Disabled.
Self-test execution status:      (   0) The previous self-test routine completed
                                   without error or no self-test has ever
```

Wyświetla szczegółowe informacje o stanie dysku z S.M.A.R.T.

```
smartctl -H /dev/sda
```

```
root@rivia:~$: smartctl -H /dev/sda
smartctl 7.4 2024-10-15 r5620 [x86_64-linux-6.14.11-4-pve] (local build)
Copyright (C) 2002-23, Bruce Allen, Christian Franke, www.smartmontools.org

=== START OF READ SMART DATA SECTION ===
SMART Status not supported: Incomplete response, ATA output registers missing
SMART overall-health self-assessment test result: PASSED
Warning: This result is based on an Attribute check.

root@rivia:~$: █
```

Szybki test zdrowia dysku (PASSED/FAILED).

LVM, RAID i ZFS

Dla systemów z LVM:

lvs # Lista logical volumes
pvs # Lista physical volumes
vgs # Lista volume groups
lvdisplay # Szczegóły LV

```
root@rivia:~# lvs
LV          VG          Attr      LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
data        pve          twi-aotz-- <130.27g          31.52  2.22
root        pve          -wi-ao---- 65.64g
swap        pve          -wi-ao---- 8.00g
vm-100-disk-0 pve        Vwi-aotz-- 5.00g data    99.64
vm-101-disk-0 pve        Vwi-aotz-- 8.00g data    52.77
vm-102-disk-0 pve        Vwi-aotz-- 8.00g data    95.10
vm-202-disk-0 pve        Vwi-aotz-- 33.00g data    73.49
vm-202-disk-1 pve        Vwi-aotz-- 4.00m data    14.06
vm-506-disk-0 pve        Vwi-a-tz-- 4.00m data    1.56
lvGLUSTERFS vgGLUSTERFS -wi-ao---- <953.87g

root@rivia:~# pvs
PV          VG          Fmt Attr PSize   PFree
/dev/sda3   pve          lvm2 a-- <222.57g 15.99g
/dev/sdb    vgGLUSTERFS lvm2 a-- <953.87g 0

root@rivia:~# vgs
VG          #PV #LV #SN Attr   VSize   VFree
pve          1  9  0 wz--n- <222.57g 15.99g
vgGLUSTERFS 1  1  0 wz--n- <953.87g 0

root@rivia:~# lvdisplay
--- Logical volume ---
LV Path                /dev/vgGLUSTERFS/lvGLUSTERFS
LV Name                 lvGLUSTERFS
VG Name                 vgGLUSTERFS
LV UUID                 GMRL13j-7iBT-0Mc3-Oryp-58XR-i87B-ft5hsQ
LV Write Access         read/write
LV Creation host, time rivia, 2024-12-09 18:46:47 +0100
LV Status                available
# open                   1
LV Size                  <953.87 GiB
Current LE               244190
Segments                 1
Allocation                inherit
Read ahead sectors       auto
- currently set to       256
Block device             252:0

--- Logical volume ---
LV Name                 data
VG Name                 pve
LV UUID                 fkPKpx-fh4D-1YKb-1dSP-hIr7-p7i2-moD19V
LV Write Access         read/write (activated read only)
LV Creation host, time proxmox, 2023-04-07 20:24:33 +0200
LV Pool metadata        data_tmeta
LV Pool data            data_tdata
LV Status                available
# open                   0
LV Size                  <130.27 GiB
Allocated pool data      31.52%
Allocated metadata       2.22%
Current LE               33348
Segments                 1
Allocation                inherit
Read ahead sectors       auto
- currently set to       256
Block device             252:6
```

Dla systemów z RAID:

```
cat /proc/mdstat # Status software RAID
mdadm --detail /dev/md0 # Szczegóły RAID array
```

Dla systemów z ZFS (FreeBSD, Linux z ZFS):

```
zpool list      # Lista pool'i ZFS
zpool status   # Status pool'i
zfs list       # Lista systemów plików ZFS
zfs get all    # Wszystkie właściwości
```

Podsumowanie praktycznych komend

```
# Quick check - co się dzieje z dyskami
df -Th
du -sh /* | sort -rh | head -10
lsblk -f

# Deep dive - znajdź największe pliki
find / -type f -size +1G -exec ls -lh {} \; 2>/dev/null

# Monitoring w czasie rzeczywistym
watch -n 2 'df -h; echo ""; du -sh /var/log /var/cache /tmp'

# Diagnoza performance
iostat -xz 2 5
iotop -o      # tylko aktywne procesy I/O
```

Revision #2

Created 23 November 2025 19:02:37 by Tomasz Konieczny

Updated 23 November 2025 20:00:24 by Tomasz Konieczny