

Title: Monitorowanie obciążenia systemu

Subject: eDokumenty - elektroniczny system obiegu dokumentów, workflow i CRM - AdminGuide/OSMonitoring

Version: 31

Date: 07/03/24 05:41:50

Table of Contents

<i>Monitorowanie obciążenia systemu</i>	3
1. Weryfikacja aktualnego obciążenia systemu przez poszczególne procesy działający w konsoli systemu Unix - aplikacja htop	3
2. Monitorowanie użycia pamięci trwałej - aplikacja iotop	3
3. Weryfikacja parametrów dysków twardych hdparm	4
4. Prosty pomiar wydajności I/O - polecenie dd	4
5. Testowanie systemu plików ZFS	5
6. Weryfikacja logów systemu	5
7. Skryptu uruchamiane z cron'a	5
8. Analiza ruchu sieciowego w Linuksie z mtr	5
9. Weryfikacja prędkości łącza za pomocą CURL	6

Monitorowanie obciążenia systemu

1. Weryfikacja aktualnego obciążenia systemu przez poszczególne procesy działający w konsoli systemu Unix - aplikacja htop

Instalacja pakietu:

```
apt-get install htop
```

Zastosowanie polecenia htop: Wyświetlane dane przedstawiają zużycie mocy obliczeniowej procesora, ilości zajmowanej pamięci RAM, czasu działania, numeru procesu itd. htop możemy użyć w zastępstwie za systemowego top'a, tylko w przypadku htop wyniki prezentowane są w sposób przejrzystszy dla użytkownika.

The screenshot shows the htop interface. At the top right, system statistics are displayed: Tasks: 132, 10 thr; 1 running; Load average: 0.67 0.79 0.89; Uptime: 38 days, 10:21:43. Below this, a bar chart shows CPU usage for 8 processors, with the first processor at 12.9%. Memory usage is shown as 2164/16085MB (13.4%) and swap usage as 889/3811MB (2.3%). The main table lists running processes with columns: PID, USER, PRI, NI, VIRT, RES, SHR, S, CPU%, MEM%, TIME+, and Command. The processes include postgres, www-data (multiple instances of apache2), memcached, and root (htop).

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
8009	www-data	20	0	314M	38936	11992	S	18.1	0.2	0:02.18	/usr/sbin/apache2 -k start
1591	postgres	20	0	3905M	194M	171M	S	6.7	1.2	0:21.41	postgres: 9.6/main: http
8230	www-data	20	0	308M	35328	17104	S	4.3	0.2	0:00.41	/usr/sbin/apache2 -k start
8259	www-data	20	0	303M	27012	11084	S	4.3	0.2	0:00.38	/usr/sbin/apache2 -k start
8101	www-data	20	0	305M	31920	16208	S	2.4	0.2	0:00.45	/usr/sbin/apache2 -k start
7009	www-data	20	0	303M	28188	12080	S	1.9	0.2	0:02.76	/usr/sbin/apache2 -k start
8005	www-data	20	0	314M	37936	11544	S	1.4	0.2	0:00.50	/usr/sbin/apache2 -k start
8234	www-data	20	0	302M	24996	10584	S	1.4	0.2	0:00.23	/usr/sbin/apache2 -k start
8232	www-data	20	0	303M	27024	12012	S	1.0	0.2	0:00.78	/usr/sbin/apache2 -k start
8308	www-data	20	0	303M	25500	10392	S	1.0	0.2	0:00.13	/usr/sbin/apache2 -k start
8100	www-data	20	0	303M	27176	11632	S	0.5	0.2	0:00.46	/usr/sbin/apache2 -k start
877	memcache	20	0	1187M	148M	2088	S	0.5	0.9	26:15.18	/usr/bin/memcached -m 4096 -I 20m -p 11211 -u memcache -l 127.0.0.1
8591	root	20	0	25812	4924	2888	R	0.5	0.0	0:00.04	htop
934	memcache	20	0	1187M	148M	2088	S	0.5	0.9	5:39.72	/usr/bin/memcached -m 4096 -I 20m -p 11211 -u memcache -l 127.0.0.1
8231	www-data	20	0	303M	26508	11316	S	0.5	0.2	0:00.27	/usr/sbin/apache2 -k start
8233	www-data	20	0	310M	34216	11056	S	0.0	0.2	0:00.37	/usr/sbin/apache2 -k start
7444	www-data	20	0	303M	26732	11572	S	0.0	0.2	0:01.73	/usr/sbin/apache2 -k start
7842	www-data	20	0	310M	35672	12912	S	0.0	0.2	0:02.06	/usr/sbin/apache2 -k start
7843	www-data	20	0	304M	27160	10712	S	0.0	0.2	0:00.43	/usr/sbin/apache2 -k start
8342	www-data	20	0	301M	24524	10648	S	0.0	0.1	0:00.17	/usr/sbin/apache2 -k start
932	memcache	20	0	1187M	148M	2088	S	0.0	0.9	5:39.28	/usr/bin/memcached -m 4096 -I 20m -p 11211 -u memcache -l 127.0.0.1
933	memcache	20	0	1187M	148M	2088	S	0.0	0.9	5:39.67	/usr/bin/memcached -m 4096 -I 20m -p 11211 -u memcache -l 127.0.0.1
935	memcache	20	0	1187M	148M	2088	S	0.0	0.9	5:39.72	/usr/bin/memcached -m 4096 -I 20m -p 11211 -u memcache -l 127.0.0.1

2. Monitorowanie użycia pamięci trwałej - aplikacja iotop

Instalacja pakietu:

```
apt-get install iotop
```

Zastosowanie polecenia iotop: Idealnym zastosowaniem dla iotop jest monitorowanie własnej aplikacji - pewnego rodzaju test, debugowanie.

Można także obserwować które aplikacje najbardziej potrzebują pamięci trwałej aby móc takim aplikacjom stworzyć specjalne miejsce w pamięci ram z szybkim dostępem.

Total DISK READ :		0.00 B/s	Total DISK WRITE :		109.82 K/s		
Actual DISK READ:		0.00 B/s	Actual DISK WRITE:		266.70 K/s		
TID	PRIO	USER	DISK READ	DISK WRITE	SWAPIN	IO>	COMMAND
28216	be/4	postgres	0.00 B/s	7.84 K/s	0.00 %	0.86 %	postgres: 9.6/main: http [local] idle
9724	be/4	postgres	0.00 B/s	7.84 K/s	0.00 %	0.28 %	postgres: 9.6/main: http [local] idle
10330	be/4	www-data	0.00 B/s	0.00 B/s	0.00 %	0.00 %	php /home/public_html/apps/backproc/thread_ignitor.php --threadId=t1 --parentId=703
702	be/4	www-data	0.00 B/s	0.00 B/s	0.00 %	0.00 %	php /home/public_html/apps/backproc/engine.php
3276	be/4	postgres	0.00 B/s	78.44 K/s	0.00 %	0.00 %	postgres: 9.6/main: checkpointer process
2416	be/4	www-data	0.00 B/s	0.00 B/s	0.00 %	0.00 %	php /home/public_html/apps/backproc/engine.php
2403	be/4	www-data	0.00 B/s	0.00 B/s	0.00 %	0.00 %	php /home/public_html/apps/backproc/engine.php
1	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	init
2	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kthreadd]
3	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ksoftirqd/0]
5	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kworker/0:0H]
7	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[rcu_sched]
8	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[rcu_bh]
9	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[migration/0]
10	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[watchdog/0]
11	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[watchdog/1]
12	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[migration/1]
13	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ksoftirqd/1]
15	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kworker/1:0H]
16	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[watchdog/2]
17	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[migration/2]
18	be/4	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[ksoftirqd/2]
531	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[xfs_mru_cache]
20	be/0	root	0.00 B/s	0.00 B/s	0.00 %	0.00 %	[kworker/2:0H]

3. Weryfikacja parametrów dysków twardych hdparm

Instalacja pakietu:

```
apt-get install hdparm
```

Polecenia wykonujemy z poziomu roota. Przykłady użycia:

Listuje informacje bezpośrednio z urządzenia (lub z '-i' z proc)

```
hdparm -I /dev/sda
```

Włącza tryb DMA dla pierwszego dysku twardego.

```
hdparm -d1 /dev/sda
```

Wyłącza tryb DMA dla pierwszego dysku twardego.

```
hdparm -d0 /dev/sda
```

Sprawdza wydajność czytania pamięć cache dla pierwszego dysku twardego.

```
hdparm -T /dev/sda
```

Sprawdza wydajność czytania pierwszego dysku twardego.

```
hdparm -t /dev/sda
```

Zastosowanie polecenia hdparm oraz przeprowadzenie testu: Połączenie dwóch parametrów -t -T otrzymamy przybliżone dane na temat wydajności naszego systemu wejścia – wyjścia (I/O) dysku twardego.

```
root@eDokumenty:/home/edokumenty# hdparm -tT /dev/sda1
/dev/sda1:
Timing cached reads: 14884 MB in 2.00 seconds = 7447.84 MB/sec
Timing buffered disk reads: 484 MB in 3.02 seconds = 160.02 MB/sec
```

Po więcej informacji odsyłamy do [artykułu](#).

4. Prosty pomiar wydajności I/O - polecenie dd

Polecenie dd jest uniksowym programem do niskopoziomowego kopiowania i konwersji surowych danych (ang. raw data). Ale może także zostać może zostać wykorzystane do prostego pomiaru wydajności I/O.

```
dd if=/dev/zero of=plik count=1 bs=1G oflag=direct
```

Zastosowanie polecenia dd: Zastosowanie poniższego polecenie spowoduje utworzenie pliku o rozmiarze 1G w lokalizacji, w której się znajdujemy. W naszym przypadku w lokalizacji `/home/edokumenty/`

```
root@eDokumenty:/home/edokumenty/# dd if=/dev/zero of=plik count=1 bs=1G oflag=direct
1+0 records in
1+0 records out
1073741824 bytes (1.1 GB) copied, 1.96308 s, 547 MB/s
```

Po więcej informacji odsyłamy do [artykułu](#).

5. Testowanie systemu plików ZFS

```
zpool iostat -v rpool
```

6. Weryfikacja logów systemu

Domyślnie logi zapisane są w `/var/log` bezpośrednio w katalogu lub z podziałem na podkatalogi z nazwą danej aplikacji.

- logi Apache: `/var/log/apache` tutaj mamy rozdzielenie na `error.log` oraz `access.log`
- logi PHP: `/var/log/php_error.log`
- logi PostgreSQL: `/var/log/postgresql/`
- logi systemowe:
 - auth - dane związane z autoryzacją
 - authpriv - inne komunikaty związane z autoryzacją
 - cron - komunikaty crona
 - daemon - inne demony
 - kern - komunikaty jądra systemu
 - mail - komunikaty związane z pocztą
 - syslog - komunikaty demona syslog
 - user - procesy użytkowników

7. Skryptu uruchamiane z cron'a

Tabeli crontab wykorzystujemy do uruchamiania skryptów np do zaplanowanych zadań. Jednym z elementów eDokumentów jest Cronrunner lub backproc które dodajemy w `/etc/crontab`, [CronnRunnerna](#) lub [backproc - synchronizacja poczty w tle](#)

Skrypty CronRunnerna możemy także wywoływać ręcznie z poziomu konsoli z parametrem ID procesu z tabeli crontab w bazie eDokumentów.

```
php -f /home/edokumenty/public_html/apps/edokumenty/CronRunner.php 6
```

Tabele crona są także wykorzystywane do uruchamiania skryptów np: backupów bazy danych, plików systemu eDokumenty. Opis skryptów backupujących uruchamianych z crona znajdują się w tym [artykule](#)

8. Analiza ruchu sieciowego w Linuksie z mtr

Instalacja pakietu: `apt-get install mtr`

My Traceroute łączy dwa narzędzie w jednym, mamy tutaj traceroute oraz ping.

Aplikację uruchamiamy poleceniem wraz z parametrem: `mtr edokumenty.eu`

Poprawny wynik prezentuje się w następujący sposób:

My traceroute [v0.85]									
eDokumenty (0.0.0.0)									
Keys: Help Display mode Restart statistics Order of fields quit									
Thu Oct 11 18:02:05 2018									
Host	Packets			Pings					
	Loss%	Snt	Last	Avg	Best	Worst	StDev		
1. 10.10.128.1	0.0%	43	0.3	0.3	0.2	0.4	0.0		
2. ip105.bnet.pl	0.0%	43	0.5	4.2	0.4	156.5	23.8		
3. do-ZSEU.zabrze.net.pl	0.0%	43	1.4	2.9	1.2	37.1	6.0		
4. z-Slask-COM.poznan-gw2-amsix.rtr.pionier.gov.pl	0.0%	42	10.7	10.3	9.5	30.8	3.2		
5. core1.ams.net.google.com	0.0%	42	27.2	27.5	27.0	34.3	1.1		
6. 108.170.241.193	0.0%	42	27.5	27.2	27.0	27.5	0.0		
7. 72.14.238.245	0.0%	42	27.4	27.2	27.1	27.4	0.0		
8. ams16s31-in-f3.1e100.net	0.0%	42	27.3	27.3	27.1	27.8	0.0		

Po więcej informacji odsyłamy do [artykułu](#).

9. Weryfikacja prędkości łącza za pomocą CURL

Instalacja pakietu:

```
apt-get install curl
```

Aplikację uruchamiamy w następujący sposób:

```
curl -s https://raw.githubusercontent.com/sivel/speedtest-cli/master/speedtest.py | python -
```

Wynik prezentuje się w następujący sposób:

```
root@eDokumenty:~# curl -s https://raw.githubusercontent.com/sivel/speedtest-cli/master/speedtest.py | python -
Retrieving speedtest.net configuration...
Testing from Betasoft Sp. z o.o. (213.5.203.160)...
Retrieving speedtest.net server list...
Selecting best server based on ping...
Hosted by Multiplay sp. z o.o. (Knurow) [11.96 km]: 41.84 ms
Testing download speed.....
Download: 811.48 Mbit/s
Testing upload speed.....
Upload: 409.25 Mbit/s
root@eDokumenty:~#
```