

Title: Monitorowanie obciążenia systemu

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

Version: 31

Date: 07/03/24 11:30:40

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. Weryfikacja logów systemu	4
6. Skryptu uruchamiane z cron'a	5

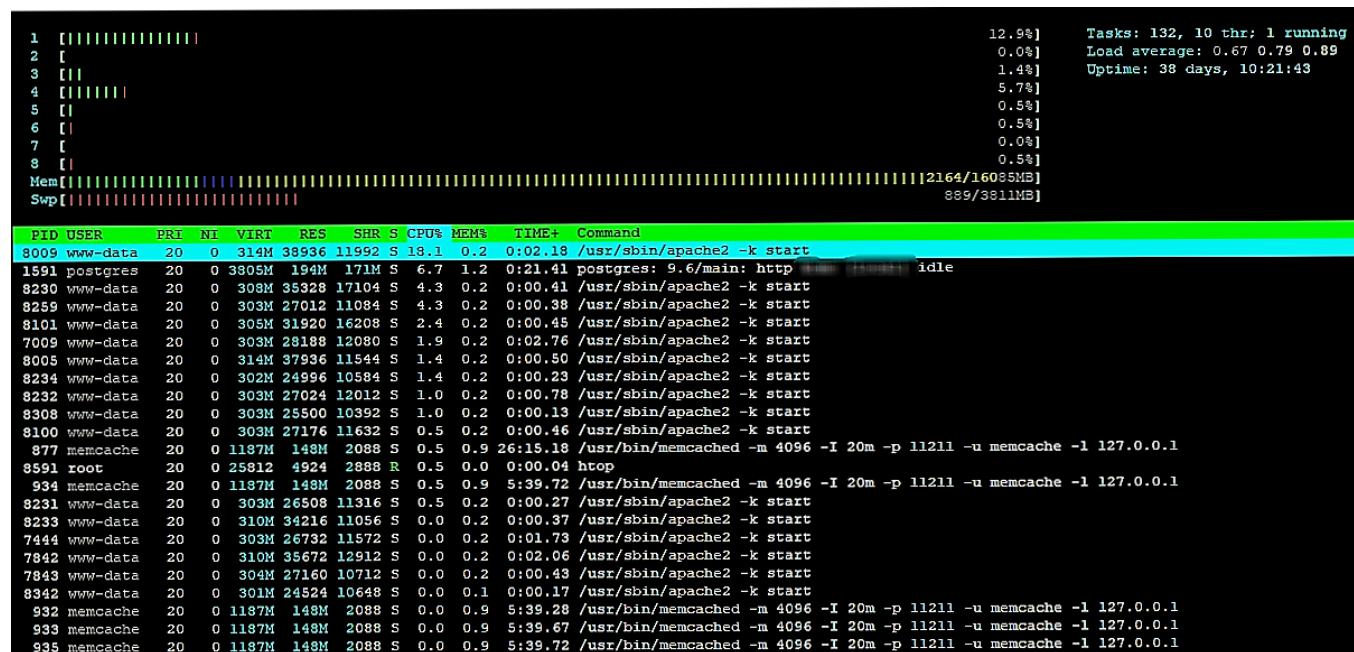
Monitorowanie obciążenia systemu

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.



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: checkpoint 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
```

4. Prosty pomiar wydajności I/O - polecenie dd

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

5. 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

6. 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, [CronnRunnera](#) lub [backproc - synchronizacja poczty w tle](#)

Skrypty CronRunnera 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: backupowych bazy danych, plików systemu eDokumenty. Opis skryptów backupujących uruchamianych z crona znajdują się w tym [artykule](#)