

TOP CPU Usage

Mar 2020

Overview

Top is a great tool for showing how much CPU and memory is being consumed on a server, on a multi CPU system however, the CPU percentages can be confusing. Look at the example below.

```
top - 10:56:52 up 3 days, 2:25, 1 user, load average: 33.93, 36.51, 34.64
Tasks: 658 total, 1 running, 657 sleeping, 0 stopped, 0 zombie
%Cpu(s): 70.8 us, 3.3 sy, 0.0 ni, 25.8 id, 0.0 wa, 0.0 hi, 0.1 si, 0.0 st
KiB Mem : 65681824 total, 47889932 free, 16411244 used, 1380648 buff/cache
KiB Swap: 0 total, 0 free, 0 used. 48418516 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
61798	ericsson	20	0	6632116	3.1g	7136	S	1377	4.9	124:42.15	liveTranscoder
61808	ericsson	20	0	6185680	2.7g	7124	S	1336	4.3	130:12.41	liveTranscoder
57234	ericsson	20	0	8333764	4.0g	7132	S	763.7	6.4	263:56.78	liveTranscoder
12224	ericsson	20	0	278752	4308	628	S	11.0	0.0	527:30.10	automation-prox
12281	ericsson	20	0	278752	4308	628	S	11.6	0.0	527:38.15	automation-prox
12637	ericsson	20	0	1134012	66676	2000	S	3.6	0.1	112:27.87	python
12302	rabbitmq	20	0	25.7g	155820	2152	S	2.6	0.2	123:23.53	beam.smp
15991	ericsson	20	0	922780	72148	2656	S	2.3	0.1	29:59.56	uwsgi
12834	ericsson	20	0	401856	26544	1808	S	1.3	0.0	33:00.22	envivio-server-
15965	ericsson	20	0	918936	68268	2776	S	1.3	0.1	15:24.63	uwsgi
12581	ericsson	20	0	1361672	58384	2032	S	1.0	0.1	34:37.70	ericsson-server
12780	ericsson	20	0	142960	4664	540	S	1.0	0.0	22:31.19	redis-server
13258	mongod	20	0	1237572	105644	7164	S	1.0	0.2	33:44.43	mongod
15944	ericsson	20	0	918816	69784	4176	S	1.0	0.1	15:18.88	uwsgi
15955	ericsson	20	0	918748	69896	4272	S	1.0	0.1	15:26.39	uwsgi
15972	ericsson	20	0	921724	71284	2852	S	1.0	0.1	15:16.50	uwsgi
15985	ericsson	20	0	922540	72272	3060	S	1.0	0.1	15:23.77	uwsgi
12303	etcd	20	0	12.7g	29472	2800	S	0.7	0.0	23:20.00	etcd
12884	ericsson	20	0	1044460	60840	2100	S	0.7	0.1	26:48.71	ericsson-interf
12890	root	20	0	406160	26488	2116	S	0.7	0.0	1:12.33	ericsson-licens
15928	ericsson	20	0	911248	65920	2812	S	0.7	0.1	0:20.61	uwsgi
15941	ericsson	20	0	918788	69968	4388	S	0.7	0.1	0:19.74	uwsgi
15960	ericsson	20	0	918836	69820	4176	S	0.7	0.1	15:17.78	uwsgi
10	root	20	0	0	0	0	S	0.3	0.0	12:15.63	rcu_sched
12237	ericsson	20	0	267176	38988	1260	S	0.3	0.1	6:45.43	celery
12261	ericsson	20	0	142964	4496	580	S	0.3	0.0	5:50.61	redis-server
12664	ericsson	20	0	758892	61888	1756	S	0.3	0.1	23:13.72	daphne
12771	ericsson	20	0	557760	30884	1920	S	0.3	0.0	6:20.14	ericsson-alarm-
12833	ericsson	20	0	142960	4484	588	S	0.3	0.0	9:07.54	redis-server

The figures for the highlighted processes are defined as 100% chunks of each CPU (or part of a CPU) so when we see a CPU usage of '1377' we know that that process is using nearly 14 CPUs worth of power (not necessarily 14 CPUs, that is different, but consuming 14 CPUs worth of resource that could be spread across more than 14 CPUs).

This makes it challenging to work out how much CPU is being used, as you need to know the number of CPUs in the first place.

Thankfully, TOP is able to show CPU usage as a percentage of total available CPU. To do this, when TOP is running press "shift - i"

```

top - 10:58:19 up 3 days, 2:26, 1 user, load average: 43.02, 38.97, 35.68
Tasks: 658 total, 2 running, 656 sleeping, 0 stopped, 0 zombie
%Cpu(s): 76.0 us, 3.6 sy, 0.0 ni, 20.4 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 65681824 total, 47763020 free, 16538076 used, 1380728 buff/cache
KiB Swap: 0 total, 0 free, 0 used. 48292068 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR  S  %CPU  %MEM     TIME+ COMMAND
 61808 ericsson  20   0 6185680    2.7g  71240 S  32.2   4.3 150:40.00 liveTranscoder
 61798 ericsson  20   0 6763188    3.1g  71360 S  25.9   5.0 142:51.28 liveTranscoder
 57234 ericsson  20   0 8464836    4.1g  71328 R  19.9   6.5 274:42.52 liveTranscoder
12281 ericsson  20   0 278752    4308    628 S   0.0   0.0  27:48.40 automation-prox
12224 ericsson  20   0 278752    4308    628 S   0.2   0.0  527:39.40 automation-prox
12302 rabbitmq  20   0   25.7g 156132   2152 S   0.1   0.2 123:26.04 beam.smp
12637 ericsson  20   0 1134012   66676   2000 S   0.1   0.1 112:30.64 python
12769 ericsson  20   0 568328   37488   2040 S   0.0   0.1 13:38.53 envivio-service
15955 ericsson  20   0 918748   69896   4272 S   0.0   0.1 15:27.09 uwsgi
15985 ericsson  20   0 922540   72272   3060 S   0.0   0.1 15:24.75 uwsgi
12780 ericsson  20   0 142960   4664    540 S   0.0   0.0 22:32.07 redis-server
12994 ericsson  20   0 17.3g   51380   2040 S   0.0   0.1 5:20.62 liveencoder-con
12261 ericsson  20   0 142964   4496    580 S   0.0   0.0 5:50.77 redis-server
12303 etcd     20   0 12.7g  29624   2800 S   0.0   0.0 23:20.62 etcd
12587 ericsson  20   0 507556   68860   1564 S   0.0   0.1 10:26.16 celery
12664 ericsson  20   0 758892   61888   1756 S   0.0   0.1 23:14.24 daphne
12834 ericsson  20   0 401856   26544   1808 S   0.0   0.0 33:01.18 envivio-server-
12884 ericsson  20   0 1044460   60840   2100 S   0.0   0.1 26:49.33 ericsson-interf
13258 mongod   20   0 1237572 105736   7164 S   0.0   0.2 33:45.48 mongod
15928 ericsson  20   0 911248   65928   2812 S   0.0   0.1 0:21.34 uwsgi
15941 ericsson  20   0 918788   69968   4388 S   0.0   0.1 0:20.43 uwsgi
15944 ericsson  20   0 918816   69784   4176 S   0.0   0.1 15:19.58 uwsgi
15960 ericsson  20   0 918836   69820   4176 S   0.0   0.1 15:18.51 uwsgi
15965 ericsson  20   0 918936   68268   2776 S   0.0   0.1 15:25.37 uwsgi
15972 ericsson  20   0 921724   71284   2852 S   0.0   0.1 15:17.27 uwsgi
15978 ericsson  20   0 921252   70864   2812 S   0.0   0.1 15:27.26 uwsgi
15991 ericsson  20   0 922780   72160   2656 S   0.0   0.1 30:00.78 uwsgi
  10 root      20   0      0      0      0 S   0.0   0.0 12:15.91 rcu_sched
11438 root      16  -4   62060    944    324 S   0.0   0.0 0:32.40 auditd

```

Above we can see that the highlighted processes now show as a total CPU consumed figure, which is much simpler to work with.

From:

<http://cameraangle.co.uk/> - WalkerWiki - wiki.alanwalker.uk

Permanent link:

http://cameraangle.co.uk/doku.php?id=wiki:top_cpu_usage

Last update: 2023/03/09 22:35

