BASH SCRIPTING

Managing Running systems

Boot, reboot, and shut down a system safely

To reboot or shutdown a Linux machine we'll often use the systemctl (system control) command.

To reboot:

```
sudo systemctl reboot

sudo systemctl start nginx

sudo systemctl stop/status mysql-server
```

To shutdown:

```
sudo systemctl poweroff
```

Rarely, you might find yourself in situations where the system refuses to reboot or shutdown normally.

```
sudo systemctl reboot --force

sudo systemctl poweroff --force
```

Scheduling Reboot

The **shutdown** command is better suited for scheduled reboots or shutdowns.

To shutdown at 02:00 AM:

```
sudo shutdown 02:00
```

The time is in 24-hour format, so you can use anything between 00:00 and 23:59.

If you want to shutdown x minutes later, use +x instead. To shutdown after 15 minutes:

```
sudo shutdown +15
```

To reboot instead, add the -r, reboot option:

```
sudo shutdown -r 02:00
sudo shutdown -r +15
sudo shutdown -r +1 'Scheduled restart to do an offline-backup of our database'
```

Diagnose and manage processes

Types of Processes:

- Foreground Processes: They run on the screen and need input from the user. For example Office Programs
- Background Processes: They run in the background and usually do not need user input. For example Antivirus.

You can use the command “fg” to continue a program which was stopped and bring it to the foreground.

```
fg jobname
```

You can use the command “&” to continue a program which was stopped and bring it to the foreground.

```
firefox bg
```

TOP

```
Processes: 497 total, 2 running, 495 sleeping, 1936 threads                                                                              08:42:55
Load Avg: 5.49, 4.65, 4.19  CPU usage: 2.58% user, 2.82% sys, 94.58% idle   SharedLibs: 350M resident, 82M data, 21M linkedit. MemRegions: 154263 total, 2250M resident, 78M private, 696M shared.
PhysMem: 7717M used (1688M wired), 474M unused. VM: 22T vsize, 3152M framework vsize, 15143833(0) swapins, 16539994(0) swapouts. Networks: packets: 13124244/5394M in, 6929299/3417M out.
Disks: 13859427/232G read, 77771389/3065G written.
```

PID	COMMAND	%CPU	TIME	#TH	#WQ	#PORTS	MEM	PURG	CMPRS	PGRP	PPID	STATE	BOOSTS	%CPU_ME	%CPU_OTHRS	UID	FAULTS	COW	MSGSENT	MSGRECV	SYSBSD	SYSMACH
9490	top	5.7	00:00.89	1/1	0	26	5692K+	0B	0B	9490	9484	running	*0[1]	0.00000	0.00000	0	4309+	96	399038+	199513+	10543+	217054+
148	WindowServer	2.5	04:14:04	14	6	3210-	337M-	7320K+	145M	148	1	sleeping	*0[1]	0.06611	0.37740	88	40431019+	156402	146655202+	106785273+	233745634+	470315309+
7673	com.apple.We	2.3	00:44.29	9	2	98	378M-	5244K	19M	7673	1	sleeping	0[10297+]	0.00000	0.00000	501	163865+	1706	99559+	67925+	268341+	317032+
9482	Terminal	1.6	00:00.85	9	4	241	28M+	6244K+	0B	9482	1	sleeping	*0[7+]	0.06634	0.03993	501	15677+	500	3679+	841+	7952+	8213+
5098	dbeaver	1.6	06:08.95	54	2	435	327M	11M	137M-	5098	1	sleeping	*1[3]	0.00000	0.00000	501	862050+	1244	321257+	41206	9024882+	7420307+
0	kernel_task	1.3	06:44:29	213/4	0	0	182M-	0B	0B	0	0	running	0[0]	0.00000	0.00000	0	488618	10365	374774300+	333724208+	0	0
74567	Google Chrom	0.9	16:02.27	32	2	3049	193M-	0B	106M-	74567	1	sleeping	*0[4469]	0.00000	0.00000	501	2317205+	33611	7647751+	4163619+	9948384+	32358994+
4697	com.apple.We	0.7	00:27.25	9	5	156	69M	336K	11M	4697	1	sleeping	*204[1]	0.00000	0.00000	501	132320+	325	83613+	37355+	479661+	284500+
133	mysqld	0.5	22:31.54	39	0	63	345M	0B	344M	133	1	sleeping	*0[1]	0.00000	0.00000	74	110630	793	547729+	273806+	71769810+	401085+
74597	Google Chrom	0.3	13:17.22	11	1	193	112M	0B	62M	74567	74567	sleeping	*1[5]	0.00000	0.00000	501	4629091	2669	5451150	2359659	2349309+	8899694+
856	fcconfig	0.3	05:12.95	5	1	55	7744K	0B	5996K	856	1	sleeping	*0[1]	0.00000	0.00000	0	96244	192	102759	5118	28658904+	1203640+
291	mds_stores	0.3	02:46:59	8	6	101+	62M+	80K	44M	291	1	sleeping	*0[1]	0.15486	0.28323	0	91626454+	6565	2571020+	2203940+	95979038+	11034040+
165	runningboard	0.3	32:30.67	7	6	470+	8284K	0B	1092K	165	1	sleeping	*6[1]	0.00000	0.23328	0	480525	148	5387238+	5269853+	216721249+	23262165+
99	mds	0.3	45:35.92	8	5	421	82M	0B	74M	99	1	sleeping	*0[1]	0.01181	0.09258	0	22146747+	314	2668520+	2302221+	97730176+	7683678+
568	mongod	0.2	13:50.03	32	0	55	17M	0B	9744K	568	1	sleeping	*0[1]	0.00000	0.00000	501	134465	420	1705057+	852503+	16345562+	1765450+
638	FortiClientA	0.1	02:50.29	9	3	1059	121M-	12K+	82M-	638	1	sleeping	*0[11783]	0.17920	0.00000	501	886205+	1062	1742387+	158215+	2679718+	3358606+
9361	mdworker_sha	0.1	00:00.09	3	1	47	1084K-	0B	0B	9361	1	sleeping	*0[1]	0.36399	0.00000	501	2418+	167	543+	240+	970+	558+
577	Siri	0.1	00:58.98	3	1	162	4668K	0B	2760K	577	1	sleeping	*0[18699+]	0.00000	0.00000	501	75085	318	623750+	96873+	335000+	1449817+
86	powerd	0.1	05:32.66	3	2	126	2024K	0B	648K	86	1	sleeping	*0[1]	0.00000	0.08945	0	932012+	164	3932415+	2170396+	5382650+	6901521+
21595	Microsoft Ex	0.1	02:43.00	22	7	571	262M	0B	238M	21595	1	sleeping	0[2159]	0.00000	0.00000	501	630574	6479	343195	120196	426518+	1064051+
4818	com.apple.We	0.1	00:04.90	16	2	239	28M	0B	14M	4818	1	sleeping	*27[2]	0.00000	0.00000	501	21463	936	61995+	26735+	103917+	79219+
637	FSMenuApp	0.1	00:15.57	5	3	232	21M-	8192B+	12M	637	1	sleeping	*0[8910]	0.03136	0.00000	501	152038	394	173580+	50879+	172786+	533758+
420	com.apple.hi	0.1	00:32.46	2	1	93	764K	0B	240K	420	1	sleeping	*0[83133+]	0.00000	0.00000	501	67410+	117	169727+	173156+	1387384+	359324+
581	GlobalProtec	0.1	01:35.91	11	5	279	24M-	592K+	17M	581	1	sleeping	*0[1]	0.03687	0.00000	501	580198	600	604101+	204658+	2590743+	3082093+
734	HuionTabletC	0.0	75:57.05	4	3	87876	15M	0B	7576K	734	1	sleeping	*0[1]	0.00000	0.00000	501	790095	204	76380557	75147837	155085992	170795807+
534	ControlCente	0.0	10:49.08	7	4	503	43M-	48K+	19M	534	1	sleeping	*0[42263]	0.04118	0.00000	501	1090727	1024	5088362+	2116754+	3120030+	7960165+
220	symptomsd	0.0	02:15.62	3	2	127+	4476K+	0B	1504K	220	1	sleeping	0[14060]	0.00000	0.00000	24	304670+	220	125731	86015	5144252+	415731
74598	Google Chrom	0.0	03:31.99	11	1	98	32M	0B	17M	74567	74567	sleeping	*0[3]	0.00000	0.00000	501	497241	2850	1465794+	1067124+	4476471+	2542525+
113	launchservic	0.0	02:49.23	4	3	534	5484K	0B	1036K	113	1	sleeping	*117[678878]	0.00000	0.00000	0	338070	142	2064503+	1494257+	4957122+	3725085+
591	TextInputMen	0.0	00:12.23	5	3	180	6860K	16K+	3336K	591	1	sleeping	*0[16068]	0.01232	0.00000	501	104202+	307	138311+	28328+	91210+	377751+
72841	Slack Helper	0.0	01:33.57	14	1	211	172M	0B	52M	72834	72834	sleeping	*0[1]	0.00000	0.00000	501	1206318	7073	184110+	108941	405769+	561625+
37545	storagekitd	0.0	00:55.79	8	4	80	3056K	0B	2424K	37545	1	sleeping	0[7]	0.00000	0.00000	0	57894	185	417414+	50628	1981795+	722209+
573	PanGPS	0.0	01:45.75	17	3	74	8996K	0B	8640K	573	1	sleeping	*0[1]	0.00000	0.00000	0	76650	305	2139	2521	5010044+	8162
1127	SiriNCServic	0.0	00:21.70	4	1	237	3832K	0B	2100K	1127	1	sleeping	*0[52980+]	0.00000	0.00000	501	110570	347	481691+	160007+	389487+	1132070+
74	logd	0.0	15:24.02	5	4	1600	18M	0B	17M	74	1	sleeping	*0[1]	0.00000	0.00000	0	2415443	124	16860799+	17924710+	44799560+	17987229+
42134	ViewBridgeAu	0.0	00:07.54	2	1	72	2040K	0B	1424K	42134	1	sleeping	*0[41285+]	0.00000	0.00000	0	19232	171	127497+	44203+	87540+	375452+
111	apsd	0.0	01:12.91	3	1	293	4420K	0B	2352K-	111	1	sleeping	*0[1]	0.14169	0.00000	0	258953+	278	458362+	201359+	1821833+	1441622+
140	notifyd	0.0	04:08.50	2	1	716	1844K	0B	320K	140	1	sleeping	*0[1]	0.00000	0.01220	0	66437	72	2575366+	2835001+	10012092+	8464793+
421	ViewBridgeAu	0.0	00:17.43	2	1	306	3108K	0B	1840K	421	1	sleeping	*0[96570+]	0.00000	0.00000	501	65216	180	299747+	108777+	227802+	885176+
37114	bluetoothd	0.0	04:27.12	10	5	294-	6736K-	0B	3020K	37114	1	sleeping	*0[1]	0.00000	0.00422	0	107835	299	1297976	946273	8620194+	2903470+
4920	com.apple.We	0.0	00:22.34	7	1	102	467M	0B	319M	4920	1	sleeping	0[2631]	0.00000	0.00000	501	246276	672	28018+	15507+	100127+	73210+
82106	Google Chrom	0.0	00:42.96	19	1	394	209M	0B	74M-	74567	74567	sleeping	*0[7]	0.00000	0.00000	501	535177+	2959	161009+	86049	290940+	398380+
5133	com.apple.We	0.0	00:48.17	9	4	98	437M	6200K	75M	5133	1	sleeping	0[13628]	0.00000	0.00000	501	213799	494	125768	73567	283552+	327857
134	cfbackd	0.0	00:22.48	11	1	89	1432K	0B	832K	134	1	sleeping	*0[1]	0.00000	0.00710	0	803058+	125	70230+	133472+	424037+	1370156+
162	contextstore	0.0	03:46.12	2	1	130-	3888K-	412K	1260K	162	1	sleeping	0[18965]	0.00000	0.00000	0	2085503	232	499411	295328	4020095+	1935028+
436	lockoutagent	0.0	00:55.48	4	3	69	1576K	0B	732K	436	1	sleeping	*0[1]	0.00000	0.00000	501	25382	153	263209+	1986	2442026+	1054529+
79	fseventsd	0.0	11:45.67	11	1	282	2640K	0B	824K	79	1	sleeping	*0[1]	0.00710	0.00000	0	2160727+	112	2607292+	857544+	34659872+	5445473+
81063	Notes	0.0	00:43.61	6	2	402	58M	0B	45M	81063	1	sleeping	0[2302]	0.00000	0.00000	501	220799	965	144992	44020	566297+	449315
211	mDNSResponde	0.0	08:07.09	3	1	218	5904K	0B	2992K	211	1	sleeping	*0[1]	0.00000	0.00000	65	486293	309	1272159	443752	19192060+	2119694
110	opendirector	0.0	05:44.19	7	6	1304-	7660K-	0B	3368K	110	1	sleeping	*0[1]	0.00000	0.00000	0	3171373	171	974521+	1097114+	16683288+	3680434+
125	dasd	0.0	08:25.01	2	1	95-	6220K-	0B	2664K	125	1	sleeping	0[53178]	0.00000	0.00000	0	347140	276	196697	168873	6396469+	780774+
177	syspolicyd	0.0	09:18.13	3	1	241	22M	0B	21M	177	1	sleeping	*0[1]	0.00000	0.00000	0	4715106	252	88102	44370	9241641+	5794288
231	mDNSResponde	0.0	00:11.77	3	1	41	1212K	0B	1148K	231	1	sleeping	*0[1]	0.00000	0.00000	0	7527	138	520287+	260194+	263035+	261427+
4695	Safari	0.0	00:43.85	8	4	712	79M	116K	25M	4695	1	sleeping	*0[292]	0.01220	0.00000	501	148138	1676	340594	149498	418967+	1124676
225	com.apple.au	0.0	00:15.99	5	2	1235	1224K	0B	524K	225	1	sleeping	0[28658]	0.00000	0.00000	202	48223	134	581546+	316113+	448759+	434752+
197	analyticsd	0.0	01:25.36	2	1	553-	4348K-	0B	1800K	197	1	sleeping	*7218-[3]	0.00000	0.00000	263	231684	167	243970	162409	1213564+	621453+
106	syslogd	0.0	01:31.90	3	2	92-	752K-	0B	240K	106	1	sleeping	0[77474]	0.00000	0.00000	0	180445	75	4628	39238	5613367+	767674

DF

```
guru99@guru99-VirtualBox:~$ df
Filesystem     1K-blocks    Used Available Use% Mounted on
/dev/sda1        7837756 2921376   4523216  40% /
udev              246488       4    246484   1% /dev
tmpfs             101512     752    100760   1% /run
none                5120       0      5120   0% /run/lock
none              253776      76    253700   1% /run/shm
```

DF

This utility reports the free disk space(Hard Disk) on all the file systems.
If you want the above information in a readable format, then use the command

`'df -h'`

FREE

This command shows the free and used memory (RAM) on the Linux system.
You can use the arguments

- free -m to display output in MB
- free -g to display output in GB

```
home@VirtualBox:~$ free
             total       used       free     shared    buffers     cached
Mem:       1026080     803604     222476          0      36312     343376
-/+ buffers/cache:     423916     602164
Swap:      1046524      35832    1010692
```

Locate and analyze system log files

System logs

System logs deal with exactly that - the Ubuntu system - as opposed to extra applications added by the user. These logs may contain information about authorizations, system daemons and system messages.

Authorization log

Location: `/var/log/auth.log`

Keeps track of authorization systems, such as password prompts, the `sudo` command and remote logins.

Daemon Log

Location: `/var/log/daemon.log`

Daemons are programs that run in the background, usually without user interaction. For example, display server, SSH sessions, printing services, bluetooth, and more.

Debug log

Location: `/var/log/debug`

Provides debugging information from the Ubuntu system and applications.

Kernel log

Location: `/var/log/kern.log`

Logs from the Linux kernel.

System log

Locate and analyze system log files

Application logs

Some applications also create logs in `/var/log`. Below are some examples.

Apache logs

Location: `/var/log/apache2/` (subdirectory)

Apache creates several log files in the `/var/log/apache2/` subdirectory. The `access.log`file records all requests made to server to access files. `error.log` records all errors thrown by the server.

X11 server logs

Location: `/var/log/Xorg.0.log`

The X11 server creates a seperate log file for each of your displays. Display numbers start at zero, so your first display (display will log to `Xorg.0.log`. The next display (display 1) would log to `Xorg.1.log`, and so on.

Non-human-readable logs

Not all log files are designed to be read by humans. Some were made to be parsed by applications. Below are some of example

Login failures log

Location: `/var/log/faillog`

Contains info about login failures. You can view it with the `faillog` command.

Last logins log

Location: `/var/log/lastlog`

Contains info about last logins. You can view it with the `lastlog` command.

Login records log

Locate and analyze system log files

Application logs
Some applications also create logs in `/var/log`. Below are some examples.

Apache logs
Location: `/var/log/apache2/` (subdirectory)

Non-human-readable logs
Not all log files are designed to be read by humans. Some were made to be parsed by applications. Below are some of examples.

Login failures log
Location: `/var/log/faillog`

Contains info about login failures. You can view it with the `faillog` command.

Last logins log
Location: `/var/log/lastlog`

Contains info about last logins. You can view it with the `lastlog` command.

Login records log
Location: `/var/log/wtmp`

Schedule tasks to run at a set date and time

Cron is a daemon used to execute scheduled commands automatically.

The commands for a cron job are stored in the crontab file on a Linux system, which is usually found in /etc/crontab. Display the contents of your crontab file with `$ crontab -l`.

Edit the crontab file with `$ crontab -e`.

```
$ <minute> <hours> <day_of_month> <month> <day_of_week> <command_to_run>
```

- minute: 0 to 59
- hours: 0 to 23
- day of the month: 1 to 31
- month: 1 to 12
- day of the week: 0 (Sunday) to 6 (Saturday)

```
# Scan my home directory for viruses

0 10 * * 1 clamscan -ir /home/don

# Backup my home directory

0 9 * * 2 tar -zcf /var/backups/home.tgz /home/don
```

Additionally, most modern cron systems feature shortcuts to common values, including:

- `@hourly` : Run once an hour (0 * * * *)
- `@daily` : Run once a day (0 0 * * *)
- `@weekly` : Run once a week (0 0 * * 0)
- `@monthly` : Run once a month (0 0 1 * *)
- `@reboot` : Run once after reboot

Assignment