

Health Check commands in Unix like OS

In this topic we will know about status of our operating system. It is very necessary to know the utilization of CPU and memory (Ram, Swap) on every Unix OS. Below mentioned some commands might be useful for every Unix system administrator.

== Top command

# top

(Display CPU usage, Memory usage, Swap Memory, Cache Size, Buffer Size, Process ID, User)

Output description:-

|  |  |  |  |
| --- | --- | --- | --- |
| PID | The process ID of each task | 1525 | 961 |
| User | The username of task owner | Home | Root |
| PR | Priority Can be 20(highest) or -20(lowest) | 20 | 20 |
| NI | The nice value of a task | 0 | 0 |
| VIRT | Virtual memory used (kb) | 1775 | 75972 |
| RES | Physical memory used (kb) | 100 | 51 |
| SHR | Shared memory used (kb) | 28 | 7952 |
| S | Status  It contains five types:            'D' = uninterruptible sleep            'R' = running            'S' = sleeping            'T' = traced or stopped            'Z' = zombie | S | R |
| %CPU | % of CPU time | 1.7 | 1.0 |
| %MEM | Physical memory used | 10 | 5.1 |
| TIME+ | Total CPU time | 5:05.34 | 2:23.42 |
| Command | Command name | Photoshop.exe | Xorg |

{ NOTE:-- If vmstat and iostat commands are not available on your box, please install sysstat package. The vmstat, sar and iostat commands are the collection of package included in sysstat }

==Sar Command :-

{SAR stands for **System Activity Report**, as its name suggest sar command is used to collect report & save CPU, Memory, I/O usage in Unix like operating system. SAR command produce the reports on the fly and can also save the reports in the log files as well}

# sar -u :-Displays CPU usage for the current day that was collected until that point. Most likely you’ll focus on the last field “%idle” to see the cpu load.

# sar -u 1 3 :-Displays real time CPU usage every 1 second for 3 times.

# sar -u ALL :-Same as “sar -u” but displays additional fields.

# sar -u ALL 1 3 :-Same as “sar -u 1 3” but displays additional fields.

# sar -f /var/log/sa/<file name> :- To display the file output.

# sar -u -f /var/log/sa/sa10 :- To displays CPU usage for the 10day of the month from the sa10 file.

# sar -n ALL :- to check the network statistics. For example: number of packets received (transmitted) through the network card, statistics of packet failure etc.

#sar -S :- to check the usage of swap memory.

== Iftop Command

(iftop is another terminal-based free open source system monitoring utility that displays a frequently updated list of network bandwidth utilization (source and destination hosts) that passing through the network interface on your system.)

First you need to install the [EPEL repository](https://www.tecmint.com/how-to-enable-epel-repository-for-rhel-centos-6-5/), :--

# yum install epel-release

# yum repolist

# yum install iftop

# iftop

== Vmstat

(To display the virtual memory statistics)

# vmstat -a

Note: If you run vmstat without parameters (means w/o -a) it’ll displays summary report since system boot.

Ouput of Vmstat:-

1. Free – Amount of free/idle memory spaces.
2. si – Swaped in every second from disk in KB.
3. so – Swaped out every second to disk in KB.

# vmstat 2 6 (vmstat execute every two seconds and stop automatically after executing six intervals.)

== IOSTAT command

(iostat without arguments displays CPU and I/O statistics of all partitions)

# iostat

Output of Iostat Command

The first section contains CPU report

%user : percentage of CPU utilization that occurred while executing at the user (application) level

%nice : percentage of CPU utilization that occurred while executing at the user level with nice priority

%system : percentage of CPU utilization that occurred while executing at the system (kernel) level

%iowait : percentage of the time that the CPU or CPUs were idle during which the system had an outstanding disk I/O request

%steal : percentage of time spent in involuntary wait by the virtual CPU or CPUs while the hypervisor was servicing another virtual processor

%idle : percentage of time that the CPU or CPUs were idle and the system did not have an outstanding disk I/O request

The second section contains device utilization report

Device : device/partition name as listed in /dev directory

tps : number of transfers per second that were issued to the device. Higher tps means the processor is busier

Blk\_read/s : show the amount of data read from the device expressed in a number of blocks (kb, mb)/s

Blk\_wrtn/s : amount of data written to the device expressed in a number of blocks (kb, mb)/s

Blk\_read : show the total number of blocks read

Blk\_wrtn : show the total number of blocks written

==> LSOF (list open files)

{the reason to use lsof command is when a disk cannot be unmounted as it says the files are being used. With the help of this command we can easily identify the files which are in use.}

# lsof |more

Ouput of LSOF Command

FD – stands for File descriptor and may seen some of the values as:

1. cwd current working directory
2. rtd root directory
3. txt program text (code and data)
4. mem memory-mapped file

Also in FD column numbers like 1u is actual file descriptor and followed by u,r,w of it’s mode as:

1. r for read access.
2. w for write access.
3. u for read and write access.

TYPE – of files and it’s identification.

1. DIR – Directory
2. REG – Regular file
3. CHR – Character special file.
4. FIFO – First In First Out

# lsof -i (list all the networks)

=> Netstat command (network statistics)

# netstat -a (List all the ports)

# netstat -l (list all active port connections)

# netstat -i (List of all network interface)

== Free command

(This command gives information about total used and available space of physical memory and swap memory.)

# free

Output is :-

* Total:- displays the total installed memory (MemTotal and SwapTotal i.e present in /proc/meminfo).
* Used:- displays the used memory.
* Free:- displays the unused memory.
* Shared:- displays the memory used by tmpfs ( i.e. present in /proc/meminfo and displays zero in case not available).
* Buffers:- displays the memory used by kernel buffers.
* Cached:- displays the memory used by the page cache and slabs(in /proc/meminfo).
* buffers/cache:- displays the sum of buffers and cache.

**Options for free command**

* -b, bytes : It displays the memory in bytes.
* -k, kilo : It displays the amount of memory in kilobytes(default).
* -m, mega : It displays the amount of memory in megabytes.
* -g, giga : It displays the amount of memory in gigabytes.
* -t, tera : It displays the amount of memory in terabytes.
* -h, human : It shows all output columns automatically scaled to shortest three digit unit and display the units also of print out. The units used are B(byte), K(kilo byte), M(mega byte), G(giga byte), and T(tera byte).
* -c, count : It displays the output c number of times and this option actually works with -s option.
* -l, low/high : It shows the detailed low and high memory statistics
* -o, old : This option disables the display of the buffer adjusted line.
* -s, seconds : This option allows you to display the output continuously after s seconds delay. In actual, the usleep system call is used for microsecond resolution delay times.
* -t, total : It adds an additional line in the output showing the column totals.
* -V, version : It displays version info and exit.

== PS command

(To check the process status on the system)

# ps

Simple ps command shows :--

PID :– the unique process ID  
TTY :– terminal type that the user is logged into  
TIME :– amount of CPU in minutes and seconds that the process has been running  
CMD :– name of the command that launched the process.

**Options for ps command**

-aux, X means process running from a “setuid” program (i.e., a process that has Effective UID ≠ Real UID) can set the EUID back to the RUID.

-eaf, to display all the running process with full format by use this option.

Output discerption :-

| Column Header | Contents |
| --- | --- |
| %CPU | How much of the CPU the process is using |
| %MEM | How much memory the process is using |
| ADDR | Memory address of the process |
| C or CP | CPU usage and scheduling information |
| COMMAND\* | Name of the process, including arguments, if any |
| NI | nice value |
| F | Flags |
| PID | Process ID number |
| PPID | ID number of the process's parent process |
| PRI | Priority of the process |
| RSS | Real memory usage |
| S or STAT | Process status code |
| START or STIME | Time when the process started |
| SZ | Virtual memory usage |
| TIME | Total CPU usage |
| TT or TTY | Terminal associated with the process |
| UID or USER | Username of the process's owner |
| WCHAN | Memory address of the event the process is waiting for |

# ps -o uid,gid -p (process id)

It will show user id and group id of the user who run the process.