|  |
| --- |
| sar Cheatsheet  Parent Category: [Documentation](https://www.bentasker.co.uk/documentation)  Published: Tuesday, 21 April 2015 04:07  Category: [Linux](https://www.bentasker.co.uk/documentation/linux)  Author: [Ben Tasker](https://www.bentasker.co.uk/about-me)  *sar* can be an incredibly helpful utility when examining system performance, but if not used regularly it's easy to forget which flags to use.  This short post details a number of useful arguments to pass    **Basic Output**  sar    **CPU Usage per Core**  sar -P ALL    **Memory Usage**  sar -r    **Swap Usage**  sar -S    **I/O**  sar -b    **I/O by Block Device**  sar -d -p    **Check Run Queue and Load Average**  sar -q    **Network Stats**  sar -n DEV  Where DEV can be one of the following   * DEV – Displays network devices vital statistics for eth0, eth1, etc., * EDEV – Display network device failure statistics * NFS – Displays NFS client activities * NFSD – Displays NFS server activities * SOCK – Displays sockets in use for IPv4 * IP – Displays IPv4 network traffic * EIP – Displays IPv4 network errors * ICMP – Displays ICMPv4 network traffic * EICMP – Displays ICMPv4 network errors * TCP – Displays TCPv4 network traffic * ETCP – Displays TCPv4 network errors * UDP – Displays UDPv4 network traffic * SOCK6, IP6, EIP6, ICMP6, UDP6 are for IPv6 * ALL – This displays all of the above information. The output will be very long.     **Historic Information**  By default, *sar* will output for the previous 24 hours, however previous days can be checked with  sar -f /var/log/sa/sa15 -n DEV |

# 2. sar CheatSheet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| OPTIONS # Live values: **interval +** **count**  **sar 1 3**   * **Each 1 second 3 line**   # historical values  **Data is captured in binary form and saved to a file**  **sar -o datafile interval count >/dev/null 2>&1 &**   |  | | --- | | All data is captured in binary form and saved to a file (datafile). The data can then be selectively displayed with the sar command using the -f option. Set the interval and count parameters to select count records at inter-val second intervals. If the count parameter is not set, all the records saved in the file will be selected. |   **Previous Days**  # Day 11 of current Month CentOS  **sar -f /var/log/sa/sa11**  **Time Range**  # show from -s [ hh:mm:ss ] to -e [hh:mm:ss]  **sar -s 10:00:00 -e 11:00:00**  **Print result with block device name**:  **sar -d -p**  **Data Options**  **sar -u # ALL CPU**  **sar -r # RAM**  **sar -b # Disk**  **sar –S # SWAP USAGE**  **sar –n DEV # DEV can be: ALL UDP TCP…**  **Mixing options, Example**   |  | | --- | | sar -b -s 10:00:00 -e 11:00:00 –f /var/log/sa/sa11  -b # disk  -s # from 10:00 to 11:00  -f # day 11  sar -r -n DEV -f /var/log/sa/sa16  Display memory and network statistics saved in daily data file  sar -I 14 -o int14.file 2 10  Report statistics on IRQ 14 for each 2 seconds. 10 lines are displayed. Data are stored in a file called int14.file. |  |  | | --- | | sar -u 1 3  **Report All CPU utilization for each 1 seconds. 3 lines are displayed**.  [root@TESTBED-VOD-CMS ~]# sar -u 1 3  Linux 2.6.32-358.el6.x86\_64 (TESTBED-VOD-CMS) 01/19/2018 \_x86\_64\_ (24 CPU)  11:21:33 AM CPU %user %nice %system %iowait %steal %idle  11:21:34 AM all 4.13 0.00 0.17 0.46 0.00 95.24  11:21:35 AM all 0.58 0.00 0.54 0.79 0.00 98.08  11:21:36 AM all 4.26 0.00 0.33 0.04 0.00 95.37  Average: all 2.99 0.00 0.35 0.43 0.00 96.23 |  |  | | --- | | sar –d –p  **Print the result with block device name**:  [root@TESTBED-VOD-CMS ~]# sar -d -p 1 3  Linux 2.6.32-358.el6.x86\_64 (TESTBED-VOD-CMS) 01/19/2018 \_x86\_64\_ (24 CPU)  12:00:25 PM DEV tps rd\_sec/s wr\_sec/s avgrq-sz avgqu-sz await svctm %util  12:00:26 PM sda 25.25 0.00 662.63 26.24 0.28 11.04 3.92 9.90  12:00:26 PM DEV tps rd\_sec/s wr\_sec/s avgrq-sz avgqu-sz await svctm %util  12:00:27 PM sda 12.12 0.00 121.21 10.00 0.12 9.83 4.00 4.85  12:00:27 PM DEV tps rd\_sec/s wr\_sec/s avgrq-sz avgqu-sz await svctm %util  12:00:28 PM sda 4.08 0.00 48.98 12.00 0.05 12.25 4.25 1.73  Average: DEV tps rd\_sec/s wr\_sec/s avgrq-sz avgqu-sz await svctm %util  Average: sda 13.85 0.00 278.38 20.10 0.15 10.80 3.98 5.51 |  Installation **CentOS**  $ sudo yum install sysstat |