How to check if server swap is a Linux / FreeBSD server performance bottleneck with vmstat

Author : admin

If you're responsible for heavily loaded bsd unix/linux based server, you will certainly be concerned about the memory consumption.
And in particular concerned that the server swap (hard disk space used as memory) is not used by the crucial system processes like: **Apache, MySQL, Squid, Qmail. etc.**

The Linux server swap issue is a real hot topic in system administration terms, however many Linux administrators are not aware of neither how SWAP memory operates on a Unix system nor how to check if system daemons/processes are using the allocated swap memory.

Just until recently I myself wasn't much informed about the topic neither and my knowledge about the Linux bsd swap system way of operation was minimal.
As I'm now in a process of performance tweakening of a Debian Linux powered server it was necessary to re-educate myself to get a better glimpse on how the swap system works and how to get statistics related to the **swap memory**

Therefore I decided to write this little article to give a very brief information on a handy command which is very helpful in server system swap monitoring and analysis.The one I mean is **vmstat**

With *vmstat*, vmstat as you can check in the command manual **Report virtual memory statistics**

The command is one of the Unix admin swiss knife;s outfit. Vmstat reports a bunch of information which is not easy to be red by amateur Unix adminsitrator however the most important information presented by vmstat consists in two returned fields, the **si** and **so** columns:

A typical output from **vmstat** looks simlar to:

```
debian:~# vmstat 5
procs -----------memory---------- ---swap-- -----io---- -system-- ----cpu----
r b swpd free buff cache si so bi bo in cs us sy id wa
3 3 641308 90468 75180 3357452 1 1 14 31 0 0 3 1 95 1
1 1 641308 60828 74900 3383536 0 0 6 524 3729 2179 4 1 94 1
0 0 641308 64176 74804 3381824 0 0 6 253 3401 2047 3 1 95 1
```

You can see from the above presented vmstat output that the information it returns looks quite obscure but easy it's actually very easy to read the major information the admin needs, the most crucial info as I've mentioned above is in the **si** and **so** columnsThe command **5** parameter instructs the vmstat program to refresh and show output from virtual memory (swap) on every 5 seconds.
The first returned output for **si and so** column fields is **1, 1** which is pretty much a default behaviour, what is more interesting is the output returned further for instance in the second returned column on the

line:

```
 r b swpd free buff cache si so bi bo in cs us sy id wa
1 641308 60828 74900 3383536 0 0 6 524 3729 2179 4 1 94 1
0 0 641308 64176 74804 3381824 0 0 6 253 3401 2047 3 1 95 1
```

In these output you can see the value of **si and so** is equal to **0, 0** which is a very perfect case and means the swap memory on this server system where I issue the vmstat command is not used by server system daemons/processes.
Nevertheless in some cases it's possible that the returned values from **vnstat** for the *si, so* fields are non zero and quite a high number if that is the case with you, this is a sure indicator the physical system memory on the server is not enough for the system processes to operate (e.g. Apache, MySQL, PostgreSQL, Squid, Qmail or whatever daemons the server runs are heavily using the server swap or in other terms the server hard disk space based memory is used for normal system operations).

Therefore what is important to be tracked is the **si, so** returned values on the Unix, Linux server does not grow much above zero r in best case stays at zero.

Now let me explain what the *si and so* abbreviations stands for. **SI** stands for the so called **page ins** **Page ins** are chunks of memories by system processes let's say apache which are sometimes stored in swap, it's a partial memory content and therefore the presence of **si's** used on a system might not be a problem.
A typical example of where **page ins** or **si**'s are used are an Apache server which does fork new proccesses to serve new clients.
in this case the **si** memory space is used just to store information about the forking of the process and after the process is once spawned the **si** is used no more.

However the **page outs** or **so** column values if present are a sure sign the SWAP is the bottleneck for bad server performance.

In that manner of thoughts if you have high value numbers for **so** is a sure sign of trouble, if you have numbers in **so** beware and start thinking about bying more memory for your server ;)

page-outs are used by the Linux system in case if the physical server memory is running low and an extra memory is needed for the server system processes. In that case the memory is freed by exporting data from physical memory to swap memory.
As you can imagine storage of processes operations is enormously slow and server system performance in case of **page-outs** / si usage is terrible!
vmstat is really handy and having a good understanding of it is very important for the Unix admin, for some further info and better depth on how useful on a unix based system it can be check the manual ;)