

How To Monitor Linux Memory Usage with Watch Command

Hi all, today I'm going to teach you not one, but two really cool things in one post! First, I'll introduce you to advanced memory usage stats available on Linux systems through `/proc/meminfo` file, and then I'll explain the basics of using the [watch command](#).

Memory usage with `/proc/meminfo`

As you know, quite a few Unix-like systems use the so-called pseudo file systems like `/proc`. It's not a real filesystem, but just a convenient representation of processes managed by your Unix OS. In Linux systems, this directory also contains quite a few files allowing you to access various information about your system. `/proc/meminfo` is one of such files, it gives you access to most of the memory usage stats.

To get a snapshot of the current state of memory usage on your Linux system, simply cat the `/proc/meminfo` file:

```
ubuntu$ cat /proc/meminfo
MemTotal:      523008 kB
MemFree:       35336 kB
Buffers:       85560 kB
Cached:        137220 kB
SwapCached:    24480 kB
Active:        327420 kB
Inactive:      91308 kB
HighTotal:     0 kB
HighFree:      0 kB
LowTotal:      523008 kB
LowFree:       35336 kB
SwapTotal:    1048568 kB
SwapFree:      998960 kB
Dirty:         504 kB
```

```
Writeback:          0 kB
Mapped:            212232 kB
Slab:              39140 kB
CommitLimit:      1310072 kB
Committed_AS:     655992 kB
PageTables:        4748 kB
VmallocTotal:    34359738367 kB
VmallocUsed:       628 kB
VmallocChunk:    34359737739 kB
```

This probably gives you more information about memory usage that you'll ever want to know, but there's quite a few really useful stats there like the MemFree or SwapFree ones, they're useful for making sure your OS environment is healthy enough in terms of having plenty of free memory for efficient operation.

Using watch command to track progress

[watch command](#) is a really neat tool which does a simple but incredibly useful thing: it repeatedly runs a given command line and shows you the output. So, you're effectively monitoring a progress of some process by watching relevant files.

The default interval is 2, which gives enough dynamics for most of the needs.

Here's how you use this command:

```
ubuntu$ watch cat /proc/meminfo
```

So it's the same command we used in previous example, `cat /proc/meminfo`, but this time we're asking the [watch command](#) to re-run the command every 2 seconds and show us the output.

The result of running a watch command is going to be a constantly refreshed console showing something like this:

```
Every 2.0s: cat /proc/meminfo          Fri Feb 13 03:51:01 2009
```

```
MemTotal:      523008 kB
MemFree:       46396 kB
Buffers:       82636 kB
Cached:        131044 kB
SwapCached:    24480 kB
Active:        308512 kB
Inactive:      99372 kB
HighTotal:     0 kB
HighFree:      0 kB
LowTotal:      523008 kB
LowFree:       46396 kB
SwapTotal:     1048568 kB
SwapFree:      998960 kB
Dirty:         832 kB
Writeback:     0 kB
Mapped:        211076 kB
Slab:          39132 kB
CommitLimit:   1310072 kB
Committed_AS: 654860 kB
PageTables:    4856 kB
VmallocTotal:  34359738367 kB
VmallocUsed:   628 kB
VmallocChunk:  34359737739 kB
```

This output gets refreshed every 2 seconds, so the numbers shown are constantly updated.

That's it for today! There are limitless possibilities for monitoring various processes using **watch command** and I'll be sure to cover them in the future, but for now – have a great weekend and hope Friday 13th turns out great!

See also:

- [How to determine physical memory size in Linux](#)
- [Unix file types](#)
- [Unix glossary](#)