

OOM killer "Out of Memory: Killed process" SOLUTIONS / SUMMARY

Source: <http://linux.derkeiler.com/Mailing-Lists/RedHat/2007-08/msg00061.html>

- *From:* Eric Sisler <esisler@xxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Fri, 10 Aug 2007 09:47:48 -0600
-

Since this problem seems to popup on different lists, this message has been cross-posted to the general Red Hat discussion list, the RHEL3 (Taroon) list and the RHEL4 (Nahant) list. My apologies for not having the time to post this summary sooner.

I would still be banging my head against this problem were it not for the generous assistance of Tom Sigtler <ttsig@xxxxxxxxxxxxxx> and Brian Long <brilong@xxxxxxxxxx>.

In general, the out of memory killer (oom-killer) begins killing processes, even on servers with large amounts (6Gb+) of RAM. In many cases people report plenty of "free" RAM and are perplexed as to why the oom-killer is whacking processes. Indications that this has happened appear in /var/log/messages:
Out of Memory: Killed process [PID] [process name].

In my case I was upgrading various VMware servers from RHEL3 / VMware GSX to RHEL4 / VMware Server. One of the virtual machines on a server with 16Gb of RAM kept getting whacked by the oom-killer. Needless to say, this was quite frustrating.

As it turns out, the problem was low memory exhaustion. Quoting Tom: "The kernel uses low memory to track allocations of all memory thus a system with 16GB of memory will use significantly more low memory than a system with 4GB, perhaps as much as 4 times. This extra pressure happens from the moment you turn the system on before you do anything at all because the kernel structures have to be sized for the potential of tracking allocations in four times as much memory."

You can check the status of low & high memory a couple of ways:

```
# egrep 'High|Low' /proc/meminfo
HighTotal: 5111780 kB
HighFree: 1172 kB
LowTotal: 795688 kB
LowFree: 16788 kB
```

```
# free -lm
```

OOM killer "Out of Memory: Killed process" SOLUTIONS / SUMMARY

```
total used free shared buffers cached
Mem: 5769 5751 17 0 8 5267
Low: 777 760 16 0 0 0
High: 4991 4990 1 0 0 0
-/+ buffers/cache: 475 5293
Swap: 4773 0 4773
```

When low memory is exhausted, it doesn't matter how much high memory is available, the oom-killer will begin whacking processes to keep the server alive.

There are a couple of solutions to this problem:

If possible, upgrade to 64-bit Linux. This is the best solution because **all** memory becomes low memory. If you run out of low memory in this case, then you're **really** out of memory. ;-)

If limited to 32-bit Linux, the best solution is to run the hugemem kernel. This kernel splits low/high memory differently, and in most cases should provide enough low memory to map high memory. In most cases this is an easy fix – simply install the hugemem kernel RPM & reboot.

If running the 32-bit hugemem kernel isn't an option either, you can try setting `/proc/sys/vm/lower_zone_protection` to a value of 250 or more. This will cause the kernel to try to be more aggressive in defending the low zone from allocating memory that could potentially be allocated in the high memory zone. As far as I know, this option isn't available until the 2.6.x kernel. Some experimentation to find the best setting for your environment will probably be necessary. You can check & set this value on the fly via:

```
# cat /proc/sys/vm/lower_zone_protection
# echo "250" > /proc/sys/vm/lower_zone_protection
```

To set this option on boot, add the following to `/etc/sysctl.conf`:

```
vm.lower_zone_protection = 250
```

As a last-ditch effort, you can disable the oom-killer. This option can cause the server to hang, so use it with extreme caution (and at your own risk)!

Check status of oom-killer:

```
# cat /proc/sys/vm/oom-kill
```

Turn oom-killer off/on:

```
# echo "0" > /proc/sys/vm/oom-kill
# echo "1" > /proc/sys/vm/oom-kill
```

To make this change take effect at boot time, add the following to `/etc/sysctl.conf`:

```
vm.oom-kill = 0
```

OOM killer "Out of Memory: Killed process" SOLUTIONS / SUMMARY

For processes that would have been killed, but weren't because the oom-killer is disabled, you'll see the following message
in /var/log/messages:

```
"Would have oom-killed but /proc/sys/vm/oom-kill is disabled"
```

Sorry for being so long-winded. I hope this helps others who have struggled with this problem.

-Eric

--

Eric Sisler <esisler@xxxxxxxxxxxxxxxxxxxxxx>
Library Network Specialist
Westminster Public Library
Westminster, CO USA

Linux - Don't fear the Penguin.
Want to know what we use Linux for?
Visit <http://wallace.westminster.lib.co.us/linux>

--

redhat-list mailing list
unsubscribe <mailto:redhat-list-request@xxxxxxxxxx?subject=unsubscribe>
<https://www.redhat.com/mailman/listinfo/redhat-list>