Table of Contents

Logbook.....1

Logbook

- Week 10

We did some pre-tests to see in what configuratie we could get the maximum performance. We made use of 4 oplapro nodes (56 to 59). First we did some Iperf tests and after that some globus-url-copy tests. These tests run well.

After these tests we started to setup our monitoring. We use Ganglia for the system monitoring and a postgress database to log grid-ftp's and to generate performance graphs.

After these tests we did some duration tests. For this we wrote a script which started globus-url-copies in the background. We tuned the script that it started 4 globus-url-copies across 2 filesystems per node. With this setup could run with about 40-50 MB/s per node. The aggregate bandwith was 180MB/s.

When this was done, on friday we started to change the setup to the Radiant load generator. The first tests failed because there as a bug in the load-generator module. And James was in a meeting. James fixed it in the weekend and run some tests.

- Week 11

I this week a lot of testing to get the setup right with the radiant server.

- Monday March 21th

Tuning issues:

During the previous weekend all rembrandt nodes were down. Rembrandt 4 and 5 were hung and 3 and 8 were not hung but lot off processes were killed and the transfers were stopped. To get a clean start I rebooted all our rembrandt nodes.

On all the rembrandt nodes in the console and the dmesg log there were a lot of messages of "swapper: page allocation failure. order:0, mode:0x20". There is not that much of information about this error and according to all available information it seems temporary. Except our nodes are hung on this. The error messages reports that the kernel is not able to allocate memory and is not able to free up memory with swapping.

I searched in the Linux/Documentation directory to see what is tunable on this. I found the following.

vm.dirty_expire_centisecs

This tunable is used to define when dirty data is old enough to be eligible for writeout by the pdflush daemons. It is expressed in 100'ths of a second. Data which has been dirty in-memory for longer than this interval will be written out next time a pdflush daemon wakes up.

This was on 3000. This means that the file data should be first cached for 30 seconds before it is eligible to be written to disk. I set this to 500 so that after 5 seconds the data can be written to disk.

vm.dirty_expire_centisecs = 500

vm.dirty_writeback_centisecs

The pdflush writeback daemons will periodically wake up and write `old' data out to disk. This tunable expresses the interval between those wakeups, in 100'ths of a second.

This was on 500. This means that every 5 seconds the pdflush daemon is waken up to see if there is dirty data in the file buffer cache to be written to disk. I set this parameter to 250 to make pdflush more agressief to write data to disk.

vm.dirty_writeback_centisecs=250

vm.dirty_ratio

Contains, as a percentage of total system memory, the number of pages at which a process which is generating disk writes will itself start writing out dirty data.

This was on 40. This means that a process must have more then 40 percent of the physical memory of data cached before it starts writing data directly to disk. This is set to 10% which makes the processess in a earlier stage responsible for writing the data to disk. In this way you less depend on the pdflush daemon on flushing the data to disk.

vm.dirty_ratio=10

lower_zone_protection

For some specialised workloads on highmem machines it is dangerous for the kernel to allow process memory to be allocated from the "lowmem" zone. This is because that memory could then be pinned via the mlock() system call, or by unavailability of swapspace.

And on large highmem machines this lack of reclaimable lowmem memory can be fatal.

So the Linux page allocator has a mechanism which prevents allocations which *could* use highmem from using too much lowmem. This means that a certain amount of lowmem is defended from the possibility of being captured into pinned user memory.

(The same argument applies to the old 16 megabyte ISA DMA region. This mechanism will also defend that region from allocations which could use highmem or lowmem).

The `lower_zone_protection' tunable determines how aggressive the kernel is in defending these lower zones. The default value is zero - no protection at all.

If you have a machine which uses highmem or ISA DMA and your applications are using mlock(), or if you are running with no swap then you probably should increase the lower_zone_protection setting.

The units of this tunable are fairly vague. It is approximately equal to "megabytes". So setting lower_zone_protection=100 will protect around 100 megabytes of the lowmem zone from user allocations. It will also make those 100 megabytes unavaliable for use by applications and by pagecache, so there is a cost.

The effects of this tunable may be observed by monitoring /proc/meminfo:LowFree. Write a single huge file and observe the point at which LowFree ceases to fall.

A reasonable value for lower_zone_protection is 100.

This was on 0 and this means no protection. I think that this is one of the main causes of the hangs in this weekend. I set this according to last line above.

vm.lower_zone_protection=100

- Monday March 21

On request of James I redefined that crontab line of the load-generator to channel NL without type 56to59. This means that we do not run on the dedicated oplapro nodes but the transfers are started via the host radiantservice.cern.ch. This host will startup the transfers on the oplapro nodes 50-79.

- Monday March 21 around 14:00

After all VM tunings rembrandt node was again in trouble. The symptom are similar as above. Only again rembrandt8 was not i a complete hang state but the kernel was starting killing processess. One of the processes was the gmond daemon of ganglia. And ganglia reported rembrandt8 down. After restarting ganglia I saw that the system load was about 30. This is pretty much for a 2 processor system. After some tops and ps'ses I saw that there were a number of ftp processess running. After killing those processes the load was decreasing to about 10, which is still high.

I also increased the vm.lower_zone_protection parameter to 200. I did this on allo the nodes. In this case the kernel has more free memory space to do its work.

I also sent an email to James to inquire about the criteria in which the load generator determines to startup a new transfer to a node. I want to limit this to a maximum of 2. If 2 transfers are running to a single receiver node the load generator should wait until 1 has finished before starting another one.

I also changed the number of file setting in the radiant server to 8.

- Monday March 21 15:26

I saw that the average load was dropped. Probably caused by the number of files (nofiles) of the radiant server (set to 8). I increased this to 16 again.

- Monday March 21 16:30

Because rembrandt8 was not function again we put the nofiles back to 8. When we looked at rembrandt8 more closely we discovered that the tuning parameters set earlier today these were not set at rembrandt8. We set them. After this rembrandt8 seems to working fine again. On rembrandt8 we set the dirty_writeback_centisecs the same as dirty_expire_centisecs on 500. This means that the pdflush daemon is waken up as fast as the pages are expired. On the other nodes the dirty_writeback_centisecs is set to 250. This means the pdflush is waken up twice as fast the pages are expired.

- Tuesday March 22 09:45

The quartet of Rembrandt nodes clearly could do with some more input. The make this happen, the "nofiles" paramenter for the NL channel was increased from 8 to 10.

- Tuesday March 22 13:10

There was a problem on the CERN side the last half hour or so. According to Mark it was something with the myproxy server. This explains the drop in throughput for the last hour. It has just been restarted.

- Tuesday March 22 15:05

The Rembrandts still find the time to do a nice ganglia painting in between. So we increment the number of files again ...

```
> [oplapro80] /opt/lcg/bin > radiant-channel-list --channel=NL --long
> #Chan : State  :    Last Active   :Bwidth: Files:   From   :   To
> NL    :Active  :05/03/22 15:04:39 :3072  :10    :cern.ch   :sara.nl
>
> [oplapro80] /opt/lcg/bin > radiant-channel-update --channel=NL --nofiles=12
```

```
> [oplapro80] /opt/lcg/bin > radiant-channel-list --channel=NL --long
> #Chan : State  :    Last Active   :Bwidth: Files:   From   :   To
> NL    :Active  :05/03/22 15:04:39 :3072  :12    :cern.ch   :sara.nl
```

- Wednesday March 23 08:00

It was a good night. During this night we got the target performance of 500MB/s from CERN. SARA reached a performance of about 90-100MB/s which was our target. To boost the performance we set the number of parallell files to 16.

- Wednesday March 23 12:00

Changed the parallel option in radiant-load-generator line in the crontab from 1 to 2 to get 2 streams per globus-url-copy. Hopefully it will boost the performance.

- Wednesday March 23 13:00

One filesystem on Rembrandt5 was overloaded. I killed some ftp sessions and changed the number of files to 8. In this should be similar as 16 single stream ftp sessions.

- Wednesday March 23 16:30

The aggregate performance dropped slightly. I changed it back from 8 nofiles dual streams to 16 nofiles single stream.

- Thursday March 24 10:00

Added a 5th node, rembrandt2, to the receiving rembrandt cluster and increased the number of files to 20.

- Thursday March 24 13:50

In the current setup we can not regulate the scheduling that it has a maximum of 4 transfers per host and spread across 2 filessystems. Sometimes we see on a host that more that 4 (up to 10) transfers and more or less on a single filesystem. This hits the disk performance and the overall performance drops.

```
UID        PID  PPID   LWP  C NLWP     SZ   RSS PSR STIME TTY          TIME CMD
sanden   13796 12860 13796  1    1 10230  4608   1 13:45 ?        00:00:06 ftpd: oplapro80-extern
sanden   13816 12860 13816  1    1 10230  4608   0 13:45 ?        00:00:07 ftpd: oplapro80-extern
sanden   13834 12860 13834  0    1 10230  4608   0 13:46 ?        00:00:04 ftpd: oplapro80-extern
sanden   13846 12860 13846  0    1 10230  4608   1 13:46 ?        00:00:04 ftpd: oplapro80-extern
sanden   13946 12860 13946  1    1 10230  4608   0 13:49 ?        00:00:05 ftpd: oplapro80-extern
sanden   13976 12860 13976  1    1 10230  4608   0 13:50 ?        00:00:05 ftpd: oplapro80-extern
sanden   13988 12860 13988  1    1 10230  4608   0 13:50 ?        00:00:03 ftpd: oplapro80-extern
sanden   14016 12860 14016  3    1 10230  4604   1 13:51 ?        00:00:06 ftpd: oplapro80-extern
sanden   14085 12860 14085  0    1 10230  4608   1 13:53 ?        00:00:00 ftpd: oplapro80-extern
sanden   14101 12860 14101  1    1 10230  4608   0 13:54 ?        00:00:00 ftpd: oplapro80-extern
sanden   14109 12860 14109  3    1 10230  4608   0 13:54 ?        00:00:01 ftpd: oplapro80-extern
```

We tuned down the number of files to 16 again.

- Thursday March 24 16:00

The load balacing is not working well, there were still a lot of transfers on a single node. Tuned down the number of files to 10 (hopefully 2 per node).

- Saterday March 26 19:30

The radiant server was not working, from 13:00h, because of a full table space. Send email to James and the problem was fixed around 19:45.

- Tuesday March 29 13:00

Everything worked well during the easter weekend but on tuesday my grid certificate was expired. And when this certificate expires the myproxy certificate does not work either. Generated new grid and myproxy certificates. The transfers where about 1/2 hour stopped.

- Wednesday March 30 14:00

Rembrandt2 given back to the UVA for other tests. We continue with 4 nodes.

- Friday April 1 8:30

Radiant database down again.

```
Can't Connect to  database : sundb07.cern.ch:pdb01-1:Lcg_radiant_prod : DBI connect('host=sundb07

Can't Connect to  database : sundb08.cern.ch:pdb01-2:Lcg_radiant_prod : DBI connect('host=sundb08

Can't open any database
```

- Friday April 1 14:15

```
Radiant database down again.

Can't Connect to  database : sundb07.cern.ch:pdb01-1:Lcg_radiant_prod : DBI connect('host=sundb07
```

-- JamesCasey - 17 Jun 2005

This topic: LCG > ServiceChallengeTwoProgressSARALogbook
Topic revision: r1 - 2005-06-17 - unknown