Subject: High Load D states..
Posted by kevinm on Sun, 08 Feb 2009 17:21:09 GMT
View Forum Message <> Reply to Message

Hi all,

Apologies for the long post, im going to try to give as much information as possible firstly.

Ive got an ongoing issue that is getting worse and worse, and was hoping that someone may be able to advise.

I have 3 HE's each with 10VE's configured, everyhting has been good for a while (months) but suddenly the following issue has started.

The load on the HE will start spiraling to over 150 ..

top - 12:04:51 up 13 days, 19:04, 10 users, load average: 157.27, 141.94, 132.80
Tasks: 560 total, 14 running, 536 sleeping, 0 stopped, 10 zombie
Cpu(s): 33.6%us, 17.5%sy, 0.0%ni, 38.2%id, 6.5%wa, 0.0%hi, 4.2%si, 0.0%st
Mem: 8162292k total, 7764320k used, 397972k free, 547924k buffers
Swap: 9896000k total, 444k used, 9895556k free, 4381104k cached

and most processes in the process table inside the VE's are going to a state of D,

ps axl | grep D yeilds :

```
5 2713030 27464 6745 16   0  27428 11172 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 29186 25527  15   0  28972 13384 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 29447 18931  15   0  27468 11692 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 29593  7108  16   0  38032 19780 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 30756 25527  15   0  27420 10816 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 30841 29673  16   0  28112 12552 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 30864 29259  16   0  27420 10796 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 30909 25527  16   0  27420 10816 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 30911 25527  16   0  27420 10856 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 30962  7108  15   0  28644 12884 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 30972  6984  16   0  27424 10816 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 30973 29673  16   0  27424 10872 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 30975 29673  15   0  28368 12800 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 30992 29259  15   0  27420 10836 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 30996 29259  15   0  32092 14128 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 31052 17075  16   0  27420 10672 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 31056 25527  15   0  29100 13556 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 31057 25527  17   0  27420 10876 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 31109  7108  16   0  27420 10672 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 31117  6984  16   0  27424 10668 -      D    ?          0:00 /usr/local/apache2/bin/httpd
```

```
5    33 31118  6984  18   0  27424 10668 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 31120 29673  16   0  27424 10752 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 31165 29259  15   0  27420 10664 -      D    ?          0:00 /usr/local/apache2/bin/httpd
5    33 31166 29259  18   0  27420 10660 -      D    ?          0:00 /usr/local/apache2/bin/httpd
```

all the entires in the WCHAN column are '-' (blank)

The disk io wait if fine and low, and system responsiveness /interactivity is good.

The boxes are debian based servers running debian lenny, and the 2.6.18-12-fza-amd64 based kernel.

All processes that are in the D state gracefully leave D, after a few seconds, however the backlog of processes and the few seconds is causing the massive load.

vmstat from when in load over 150

```
procs -----------memory---------- ---swap-- -----io---- -system-- ----cpu----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa
17  1    444 196608 521836 4159924    0    0   142    98    2    2 36 17 36 11
 9  0    444 160488 522116 4163716    0    0  1766     0 3800 7196 27 18 47  8
11  2    444 187676 522388 4164976    0    0  1112   750 6193 9430 26 16 49  9
 6  0    444 149924 522468 4166180    0    0   538     0 3516 5334 38 14 45  3
17  3    444 127716 522588 4168444    0    0  1056   810 3723 5428 24 15 58  3
 7  2    444 125796 522164 4156544    0    0  1944     0 3779 4571 26 22 44  9
71 11    444 211116 522436 4159740    0    0  1246     0 5342 9873 37 37 17  8
31  5    444 259436 522592 4161532    0    0  1090  1242 5498 7995 38 53  6  3
 7  0    444 279044 522688 4163248    0    0   982     0 3874 5407 24 15 55  6
12  0    444 248580 522980 4165340    0    0   770   940 3674 5424 26 20 47  8
19  0    444 250328 523084 4165304    0    0   434     0 3517 4358 19 20 57  3
15  1    444 227456 523300 4166840    0    0   658     0 4173 6153 34 35 29  3
 7  3    444 201040 523556 4169796    0    0  1208  1398 3739 5828 35 27 27 12
12  1    444 223796 523800 4172304    0    0   786     0 2900 4334 22 20 40 18
12  1    444 228464 524020 4174836    0    0  1426  1236 3978 6281 22 14 48 16
18  1    444 238912 524100 4175368    0    0   280     0 3422 5262 21 19 50 10
14  1    444 218104 524412 4174232    0    0   480     0 4772 7138 35 27 33  5
16  2    444 212400 524892 4176500    0    0  1706   748 5724 8704 27 31 35  8
20  3    444 143560 525244 4180108    0    0  1562  1140 5482 8092 41 31 19 10
14  1    444 167844 525644 4182584    0    0  1422  1904 4559 8355 26 18 39 16
11  2    444 154960 525836 4184832    0    0   940    54 4464 7077 32 29 35  4
```

I have isolated , and firewalled a VE off, and then set apache to start a single child process, then started stracing this single process to see which syscall is causing the slow down, this was successful, in a VE with no other activity running due to the firewall I could reproduce the D state with a simple hit to the apache server status page to 127.0.0.1

time lynx --dump http://127.0.0.1/server-status123

yields

real 0m4.087s
user 0m0.007s
sys 0m0.003s

and the strace shows

open("/home/vol4/.htaccess", O_RDONLY|O_LARGEFILE) = -1 ENOENT (No such file or directory)
lstat64("/home/vol4/b/.htaccess", {st_mode=S_IFDIR|0755, st_size=48, ...}) = 0
capget(0x19980330, 0, {CAP_DAC_OVERRIDE,
CAP_DAC_OVERRIDE|CAP_SETGID|CAP_SETUID, 0}) = 0
capset(0x19980330, 0, {CAP_DAC_OVERRIDE|CAP_SETGID|CAP_SETUID,
CAP_DAC_OVERRIDE|CAP_SETGID|CAP_SETUID, 0}) = 0
setgid32(65534) = 0
setuid32(77777777)

at this point here

setuid32(77777777) there is a 2-3 second pause before continuing to print

= 0
capget(0x19980330, 0, {CAP_DAC_OVERRIDE|CAP_SETGID|CAP_SETUID,
CAP_DAC_OVERRIDE|CAP_SETGID|CAP_SETUID, 0}) = 0
capset(0x19980330, 0, {0, CAP_DAC_OVERRIDE|CAP_SETGID|CAP_SETUID, 0}) = 0

and the rest of the sys calls. Obviously it seems like the setuid that is occurring is having some performance issues.

The setuid is called using posix libcap functionality.

removing the setuid from the process causes

time lynx --dump http://127.0.0.1/server-status123

real 0m0.029s
user 0m0.007s
sys 0m0.002s

which is 'as expected' , however the setuid is required

userbean counters show that no counters are being breached, and cpu usage looks good.

This has now been occurring for over 5 days, and I have tried most things that I can think of any advise / suggestions would be MOST appreciated.

Best Regards
Kev

Subject: Re: High Load D states..
Posted by maratrus on Wed, 11 Feb 2009 12:59:50 GMT
View Forum Message <> Reply to Message

Hello,

Quote:
removing the setuid from the process causes

Does removing setuid solve the problem in general?

Quote:
setgid32(65534) = 0
setuid32(77777777)

Does setuid32 function always have "77777777" as a parameter? Or this number always changes? Is 77777777 a valid number?

Have you got a chance to reproduce this situation and look at fluctuation of uid_cache in /proc/slabinfo output?

Subject: Re: High Load D states..
Posted by kevinm on Thu, 12 Feb 2009 16:56:11 GMT
View Forum Message <> Reply to Message

Hi there,

Thanks for the reply !

Quote:Does removing setuid solve the problem in general?

Yes, removing the setuid from the process removes the 'wait' states introduced by this syscall / the delay in the syscall and solves the problem (however removes a large security layer)

Quote:Does setuid32 function always have "77777777" as a parameter? Or this number always changes? Is 77777777 a valid number?

This number always changes with the user calling the script from the web server, and the number is valid (the uid is stat'd from the file system, however in the strace the stat is completed, and the number displayed BEFORE the delay, the delay occurs when doing the setuid)

The issue shows itself when the system is under a high request per second schenario (peak points of the day, however is fully reproducable , I will see if I can get some infor from salbinfo when this is occuring.

Is there any tunables that you are aware of for the uid_cache that I can look at ?

Again THANKS for the reply, this is causing me a lot of troubles.

Regards
Kev

Subject: Re: High Load D states..
Posted by maratrus on Thu, 12 Feb 2009 17:27:57 GMT
View Forum Message <> Reply to Message

Hello,

one more question please.
May be not related to the problem but
how many users do you have inside problem VE?

Subject: Re: High Load D states..
Posted by kevinm on Thu, 12 Feb 2009 17:52:56 GMT
View Forum Message <> Reply to Message

The users are 'virtual' users, so do not have /etc/passwd entries, etc , they are simply uid's set on the file system, the apache config then runs the setuid that then sets the uid of the active process to the uid of the owner of the file called using libcap posix uid switching.

The HE runs 10 VE's , and each VE at the point of breaching would be processing some 50-60 active connections / requests per second or so, in total there would be some 500-700 active connections / setuid calls per second at the point.

In total we have over 200,000 uid's however the numbers that are being in use / setuid'ing at any one time is ased on the traffic / sites that are active.

I managed to reduce the impact of the setuid call slowness, by pushing keepalives on apache rather high, which causes more requests to be served in the one uid set, this is a 'patch' at the best of times as it exposes the apache services to dos due to apache starving.

again a big thank you for your interest.

Best Regards
Kev

Subject: Re: High Load D states..
Posted by maratrus on Fri, 13 Feb 2009 10:08:18 GMT
View Forum Message <> Reply to Message

Hello,

I haven't been able to reproduce this issue locally.
Do you have a chance to provide me with problem VE tarball as well as the steps have to be done to reproduce the problem?
I realize that this might be absolutely inconvenient for you but may be you have a chance to reproduce the problem on a newly created VE or at least can provide the detailed description what configuration file should be used for VE's web server and what kind of scripts should it invoke to get such behavior. Many thanks for your assistance.

Subject: Re: High Load D states..
Posted by kevinm on Fri, 13 Feb 2009 13:29:56 GMT
View Forum Message <> Reply to Message

I think it may be difficult to reproduce in an external environment, due to the 'stress' that may be occurring with the live traffic in my environment.

Would ssh access maybe be helpful ?

Kev

Subject: Re: High Load D states..
Posted by kevinm on Fri, 13 Feb 2009 13:42:49 GMT

View Forum Message <> Reply to Message

I think it may be difficult to reproduce in an external environment, due to the 'stress' that may be occurring with the live traffic in my environment.

Would ssh access maybe be helpful ?

Kev

Subject: Re: High Load D states..
Posted by maratrus on Fri, 13 Feb 2009 13:50:26 GMT
View Forum Message <> Reply to Message

Unfortunately, I have no explicit hypothesis right now.
So, I was going to play a little bit with kernel and with problem VE and suppose ssh connection is not required right now but I'll keep it in mind.
Quote:
I think it may be difficult to reproduce in an external environment

But if I'm not mistaken you said that you had been able to reproduce it. So may be it worth trying.

BTW what about uid_cache dynamics?

Subject: Re: High Load D states..
Posted by kevinm on Fri, 13 Feb 2009 14:07:04 GMT
View Forum Message <> Reply to Message

Quote:But if I'm not mistaken you said that you had been able to reproduce it. So may be it worth trying.

Correct it is reproducible in my environment, however with real live users using the live system, the workload may be hard to reproduce in a lab .

Quote:BTW what about uid_cache dynamics?
I will at peak traffic point today remove the keepalive apache patches, and monitor the uid_cache dynamics while it is occurring and get a log .

Thanks for your time in this matter !

Kev

Subject: Re: High Load D states..
Posted by kevinm on Fri, 13 Feb 2009 16:21:05 GMT

View Forum Message <> Reply to Message

Hi there,

Ok I have reproduced this while taking a snapshot of the slabinfo , and specifically the uid_cache entries in the slabinfo

This is a capture every second of /proc/slabinfo |grep uid_cache every second while running undre heavy keepalives to reduce the setuid call numbers / reduce the imact of the issue :

http://pastebin.com/m7584aef7

and this is a capture of the slabinfo with apache keepalievs disabled (thus causing many more setuid's per second)

http://pastebin.com/m4584e50b

Please let me know if this is as needed / does this show that the uid cache is being exhausted ?

Best Regards
Kev

Subject: Re: High Load D states..
Posted by kevinm on Fri, 13 Feb 2009 20:27:42 GMT
View Forum Message <> Reply to Message

Hi ,

thanks for your clue Marat I will surely in future throughly examine the slab allocations before most things now.

Thanks for the assistance .

Kev