

DragonFlyBSD - Bug #1984

hammer mount fails after crash - HAMMER: FIFO record bad head signature ..

02/08/2011 10:23 PM - thomas.nikolajsen

Status:	New	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
Description			
<p>On master from February 6th, I had a system freeze during 'hammer cleanup' on a x86_64 system using SMP kernel. System stopped responding on console and network (e.g. escape to debugger didnt work).</p> <p>After power cycling boot fails on mount of this root hammer file system.</p> <p>Booting from another partition (i386) R/W mount of affected hammer file system fails, but R/O mount succeeds, see below.</p> <p>I doesn't look like a disk error as it is quite new and I haven't experienced errors reading or writing to this disk.</p> <p>This file system doesn't contain any valuable data.</p> <p>-thomas</p> <p>- R/W mount fails: HAMMER(ROOT64) recovery check seqno=008f373d HAMMER(ROOT64) recovery range 3000000000242a80-30000000001c9a60 HAMMER(ROOT64) recovery nexto 30000000001c9a60 endseqno=00990a0a HAMMER(ROOT64) recovery undo 3000000000242a80-30000000001c9a60 (108556256 bytes)(RW) HAMMER(ROOT64) Found REDO_SYNC 3000000000159958 HAMMER(ROOT64) Ignoring extra REDO_SYNC records in UNDO/REDO FIFO. HAMMER(ROOT64) Ignoring extra REDO_SYNC records in UNDO/REDO FIFO. HAMMER(ROOT64) recovery complete HAMMER(ROOT64) recovery redo 3000000000242a80-30000000001c9a60 (108556256 bytes)(RW) HAMMER(ROOT64) Embedded extended redo 3000000000159958, -108097240 extbytes HAMMER: FIFO record bad head signature a733 at 3000000000159958 HAMMER(ROOT64) Illegal UNDO TAIL signature at 3000000000159958 HAMMER(ROOT64) End redo recovery</p> <p>- R/O mount succeeds: HAMMER(ROOT64) recovery check seqno=008f373d HAMMER(ROOT64) recovery range 3000000000242a80-30000000001c9a60 HAMMER(ROOT64) recovery nexto 30000000001c9a60 endseqno=00990a0a HAMMER(ROOT64) recovery undo 3000000000242a80-30000000001c9a60 (108556256 bytes) (RO) HAMMER(ROOT64) Found REDO_SYNC 3000000000159958 HAMMER(ROOT64) Ignoring extra REDO_SYNC records in UNDO/REDO FIFO. HAMMER(ROOT64) Ignoring extra REDO_SYNC records in UNDO/REDO FIFO. HAMMER(ROOT64) recovery complete HAMMER: recovered aliased 800000037eeac000 HAMMER: recovered aliased 800000037ebc8000 HAMMER: recovered aliased 800000037eb94000 HAMMER: recovered aliased 800000037eb94000</p>			

History

#1 - 02/08/2011 11:12 PM - dillon

:New submission from Thomas Nikolajsen <thomas.nikolajsen@mail.dk>:

:
:On master from February 6th, I had a system freeze during 'hammer cleanup'
:on a x86_64 system using SMP kernel. System stopped responding on console and
:network (e.g. escape to debugger didnt work).

It's definitely a software bug somewhere. How big is the filesystem?
The REDO range is 108MB and might have underflowed the undo/redu FIFO,
which is a condition I check for but which I've never been able to test.

The filesystem itself is probably fine, an inability to run REDOs doesn't
mess anything up (it just means the REDOs couldn't be run). So I think
we can get the R/W mount working again by changing the fatal error to
a non-fatal error:

fetch <http://apollo.backplane.com/DFlyMisc/hammer26.patch>

:This file system doesn't contain any valuable data.

:
:-thomas

If possible before you wipe the filesystem and BEFORE you do a R/W
mount, could you run the following command on it and redirect the output
to a file and throw it onto leaf? That may help me figure out what is
going on.

```
hammer -f <device-from-fstab> show-undo
```

Once you've done that I would appreciate it if you could try a kernel
w/ the above specified patch, see if you can mount the filesystem R+W,
and continue using it.

-Matt

#2 - 02/09/2011 09:58 PM - thomas.nikolajsen

The file system is 20GB.

I know this is rather small for a HAMMER FS.
It is just a root file system, for a x86_64 setup,
had to 'steal' from swap partition (still have 12GB for 8 GB mem)
it is not full at all.
Disklabel was already setup with i386 DragonFly system;
btw setting up dual boot i386/x86_64 works out quite easily w/ dloader ;-)

show-undo output is put on leaf:
<http://leaf.dragonflybsd.org/~thomas/issue1984>

Using supplied patch file system mounts R/W;
initially i mounted from i386 DragonFly, it seems fine.

After that I installed kernel w/ patch on x86_64 system,
it also mounted FS (now as root);
but after running for a few seconds it started giving errors;
the hammer_del_buffers message seemed endless; I had to power cycle.

I have no immediate plans to reformat FS; so if you have more ideas
on how to fix this I am all ears.

-thomas

```
-  
Feb 9 21:58:32 octopus kernel: tryroot sermo/S1VZJ90SB10754.s4d  
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) recovery check seqno=008f373d  
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) recovery range 3000000000242a80-  
30000000001c9a60  
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) recovery nexto 30000000001c9a60  
endseqno=00990a0a  
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) recovery undo 3000000000242a80-  
30000000001c9a60 (108556256 bytes)(RW)  
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) Found REDO_SYNC 3000000000159958  
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) Ignoring extra REDO_SYNC records  
in UNDO/REDO FIFO.  
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) Ignoring extra REDO_SYNC records  
in UNDO/REDO FIFO.  
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) recovery complete  
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) recovery redo 3000000000242a80-
```

3000000001c9a60 (108556256 bytes)(RW)
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) Embedded extended redo
300000000159958, -108097240 extbytes
Feb 9 21:58:32 octopus kernel: HAMMER: FIFO record bad head signature a733 at
300000000159958
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) Illegal UNDO TAIL signature at
300000000159958
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT64) End redo recovery
Feb 9 21:58:32 octopus kernel: HAMMER: Ignoring errors from REDO scan and
allowing R/W mount
Feb 9 21:58:32 octopus kernel: Mounting devfs
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT) recovery check seqno=009165d5
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT) recovery range 300000000c9974c0-
300000000c9974c0
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT) recovery nexto 300000000c9974c0
endseqno=009165d6
Feb 9 21:58:32 octopus kernel: HAMMER(ROOT) mounted clean, no recovery needed
Feb 9 21:58:32 octopus kernel: HAMMER: Warning: UNDO area too small!
Feb 9 21:58:32 octopus kernel: HAMMER: Warning: UNDO area too small!
..
Feb 9 21:00:23 octopus kernel: hammer_del_buffers: unable to invalidate
80000002b8cc4000 buffer=0xfffffe0873128e8 rep=1
Feb 9 21:00:23 octopus kernel: hammer_del_buffers: unable to invalidate
80000002b8cc8000 buffer=0xfffffe08955a778 rep=1
Feb 9 21:00:23 octopus kernel: hammer_del_buffers: unable to invalidate
80000002b8ccc000 buffer=0xfffffe08955bb28 rep=1
Feb 9 21:00:23 octopus kernel: hammer_del_buffers: unable to invalidate
80000002b8cd0000 buffer=0xfffffe087b2da28 rep=1
Feb 9 21:00:23 octopus kernel: hammer_del_buffers: unable to invalidate
80000002b8cd4000 buffer=0xfffffe08955c2a8 rep=1

#3 - 02/10/2011 03:18 AM - dillon

:Thomas Nikolajsen <thomas.nikolajsen@mail.dk> added the comment:
:
:
:The file system is 20GB.
:
:
:I know this is rather small for a HAMMER FS.
:It is just a root file system, for a x86_64 setup,
:had to 'steal' from swap partition (still have 12GB for 8 GB mem)
:it is not full at all.
:Disklabel was already setup with i386 DragonFly system;
:btw setting up dual boot i386/x86_64 works out quite easily w/ dloader ;-)
:
:show-undo output is put on leaf:
:<http://leaf.dragonflybsd.org/~thomas/issue1984>
:
:Using supplied patch file system mounts R/W;
:initially i mounted from i386 DragonFly, it seems fine.
:
:After that I installed kernel w/ patch on x86_64 system,
:it also mounted FS (now as root);
:but after running for a few seconds it started giving errors;
:the hammer_del_buffers message seemed endless; I had to power cycle.
:
:I have no immediate plans to reformat FS; so if you have more ideas
:on how to fix this I am all ears.
:
:-thomas

How is it after the power cycle? Is it still throwing errors?

I'm probably not flushing the undo buffers out of the buffer cache
in the error path for this particular error, and if that is the
case it should be possible to mount it R+W, sync, umount, and remount
R+W again and the messages should go away.

-Matt

#4 - 02/10/2011 03:30 AM - dillon

:Thomas Nikolajsen <thomas.nikolajsen@mail.dk> added the comment:
:
:
:The file system is 20GB.
:
:

:I know this is rather small for a HAMMER FS.
:It is just a root file system, for a x86_64 setup,
:had to 'steal' from swap partition (still have 12GB for 8 GB mem)
:it is not full at all.
:Disklabel was already setup with i386 DragonFly system;
:btw setting up dual boot i386/x86_64 works out quite easily w/ dloader ;-)
:
:show-undo output is put on leaf:
:http://leaf.dragonflybsd.org/~thomas/issue1984
:
:Feb 9 21:58:32 octopus kernel: HAMMER: Warning: UNDO area too small!
:Feb 9 21:58:32 octopus kernel: HAMMER: Warning: UNDO area too small!
:..
:priority: -> bug

Ok, I'm fairly certain that it is an UNDO/REDO FIFO overflow due to the mechanics of how HAMMER operates when this warning is active.

Right now when HAMMER is forced to do mini-flushes inside the main flush due to the UNDO area being too small it still doesn't flush the volume header until the more encompassing meta-flush is done. I'm certain this is causing the FIFO to overflow and blowing up the recovery code.

A 20GB HAMMER filesystem only reserves a 100MB UNDO/REDO FIFO. Even a 200GB HAMMER filesystem only reserves a 232MB UNDO/REDO FIFO. The real problem here is that the required size for the UNDO/REDO FIFO is related more to the system's ram and buffer cache configuration than to the filesystem size. I think I'm going to have to change newfs_hammer to create a minimum 500MB UNDO/REDO FIFO.

I will also have to change the flush mechanics to avoid the mini-flushes in the first place.

-Matt
Matthew Dillon
<dillon@backplane.com>

#5 - 02/10/2011 09:20 PM - thomas.nikolajsen

I tried mounting (R/W) FS from i386 kernel, like I did after applying patch, it mounts clean, without errors or redos.

Then I booted x86_64 kernel using FS as root, here I got the hammer_del_buffers error stream after some job (periodic daily) started using the FS.

After that I set hw.physmem=1G as you mentioned that too much RAM could trigger problem; this seemed to help, as it didn't error out, and after rebooting normally (without limiting used memory) haven't given any errors; haven't seen any signs of problem in FS; e.g. tried mirror-read of all PFSs, as I seem to remember that you earlier has mentioned that this will check data in PFS.

Enlarging UNDO/REDO buffer to 500MB seems like a good idea, if it will help making HAMMER more stable; this is 1% of 50GB, minimum recommended FS size, which doesn't seem like a high overhead.

-thomas

#6 - 02/10/2011 09:36 PM - dillon

:Thomas Nikolajsen <thomas.nikolajsen@mail.dk> added the comment:

:
:I tried mounting (R/W) FS from i386 kernel, like I did after applying patch,
:it mounts clean, without errors or redos.
:
:Then I booted x86_64 kernel using FS as root, here I got the hammer_del_buffers
:error stream after some job (periodic daily) started using the FS.

Make sure you are using the absolute latest master for x86-64. As was mentioned there was a bug where the physmem calculation got completely broken. It should be properly fixed now as of

39d69daecef529eb49d36fefa429c8ac08e7cbc1 and
7a3eee88d3ffab887e1b2d812672f20071d39947

You shouldn't need any memory restrictions any more.

-Matt
Matthew Dillon
<dillon@backplane.com>

#7 - 02/16/2011 11:43 PM - thomas.nikolajsen

Thanks for the heads up on hw.physmen; I used fresh master, so its OK.
I haven't seen any further problems using the FS; no crashes either.

To enlarge UNDO/REDO FIFO on FS I need to newfs_hammer FS, right?
(of cause backup data before newfs_hammer ;-)

If I understand, UNDO/REDO FIFO is only in first volume,
also called root volume, in HAMMER FS, right?

If so, it could be an idea to ask user for planned size of FS
when doing newfs_hammer (new option);
he might use a small root volume (e.g. 50GB), and later add some big volumes.

Is UNDO/REDO FIFO of 0.1% total FS size recommended for a big HAMMER FS,
or how large should it be depending on FS size and RAM size?
(maybe size also depends on I/O bandwidth of disk subsystem, if e.g.
a given number of seconds worth of disk I/O should be in UNDO/REDO FIFO)

Do you plan changes to HAMMER FS / VFS for things like this issue?

-thomas

#8 - 03/08/2011 06:57 PM - thomas.nikolajsen

I did a newfs_hammer (and backup / restore)
to get a bigger UNDO/REDO FIFO.

Is more work on this issue planned?
Otherwise I will just close it.

-thomas