

FreeNAS - Bug #25905

ZFS Send Failing To Copy After First Snapshot

09/18/2017 08:14 AM - Dan Boothby

Status: Closed: Cannot reproduce	
Priority: Important	
Assignee: Alexander Motin	
Category: OS	
Target version: N/A	
Seen in: 11.0-U3	Needs Merging: Yes
Severity: New	Needs Automation: No
Reason for Closing:	Support Suite Ticket: n/a
Reason for Blocked:	Hardware Configuration: Two zpools
Needs QA: Yes	ChangeLog Required: No
Needs Doc: Yes	
Description	
<p>I'm trying to migrate all my data from one zpool (Tank) to a new one (Ark), which are both on the same machine, via ZFS send. I've created a recursive snapshot of the whole pool Tank, and am sending it to Ark via the following command:</p> <pre>zfs send -Rv Tank@datamigrate-20170913 zfs receive -Fds Ark</pre> <p>This starts copying until it finishes the first dataset's snapshot at which point it errors and stops the transfer:</p> <pre>... 22:21:15 5.45T Tank@datamigrate-20170913 internal error: Invalid argument warning: cannot send 'Tank@datamigrate-20170913': signal received Abort (core dumped)</pre> <p>Is this a bug, or have I done something wrong with my zfs send/receive flags?</p> <p>I ran into this issue on 9.10.2-U6, but it has persisted since upgrading to 11.0-U3.</p> <p>I have tried:</p> <ul style="list-style-type: none">- Updating the zfs feature flags from the 11.0 update- Deleting the zpool Ark and recreating it under 11.0- Deleting the snapshot and recreating it under 11.0- Trying with a different snapshot- Trying with a smaller dataset	
Related issues:	
Related to FreeNAS - Bug #24732: The replication failed for the local ZFS vol...	Closed: Not To Be Fixed

History

#1 - 09/18/2017 08:44 AM - Dru Lavigne

- Status changed from Unscreened to 15

Dan: please attach a debug (System -> Advanced -> Save Debug). We'll mark the ticket as private until the dev has a chance to review the debug.

#2 - 09/18/2017 09:41 AM - Dan Boothby

- File debug-freenas-20170918174010.tgz added

Debug attached. It was maybe a week ago that I last tried the zfs send operation, so if it's not showing, I can run another and upload a new log.

#3 - 09/18/2017 09:53 AM - Dru Lavigne

- Status changed from 15 to Unscreened
- Assignee changed from Release Council to Sean Fagan
- Private changed from No to Yes

Sean: please indicate if a newer debug is needed once you have had a chance to review this one.

#4 - 09/18/2017 10:56 AM - Sean Fagan

- Status changed from Unscreened to 15
- Priority changed from No priority to Expected
- Target version set to 11.1

I can't tell what's going on here. It's doing an abort, but not logging anything, and the debug-collection doesn't include core dumps.

There should be a file, var/db/system/cores/zfs.core, which you could attach. However, equally helpful would be if you could run the 'zfs receive' with -v, just to see what it prints out.

#5 - 09/18/2017 11:58 AM - Dan Boothby

- File zfs.core added

I'm afraid it's the exact same error with the -v flag on the zfs receive side.

I've attached the zfs.core file!

#6 - 09/18/2017 12:18 PM - Sean Fagan

- Assignee changed from Sean Fagan to Alexander Motin

```
#0 0x0000000801376d0a in __rallocm (ptr=0x18f79, rsize=0x6, size=0, extra=34380148010, flags=0)
at /usr/obj/usr/src/lib/libc/jemalloc_jemalloc.c:2710
#1 0x0000000801376c49 in __allocm (ptr=0x619680, rsize=0x80109dcfb, size=<value optimized out>,
flags=<value optimized out>) at arena.h:1380
#2 0x0000000801083143 in zfs_standard_error_fmt (hdl=0x8fffffff, error=<value optimized out>,
fmt=0x16 <Address 0x16 out of bounds>) at libzfs_util.c:277
#3 0x0000000801082af5 in propname_match (p=0x802e40000 "\026\b", len=24, prop_entry=<value optimized out>)
at zprop_common.c:223
#4 0x00000008010788c7 in zfs_receive_impl (hdl=<value optimized out>, tosnap=<value optimized out>,
originsnap=0x802e8a040 "\r", flags=<value optimized out>, infd=-1, props=<value optimized out>,
sendfs=<value optimized out>) at libzfs_sendrecv.c:2823
#5 0x0000000801078b48 in zfs_receive_impl (hdl=<value optimized out>, tosnap=<value optimized out>,
originsnap=0x802e32600 "&?\002\b", flags=<value optimized out>, infd=0, props=<value optimized out>,
sendfs=<value optimized out>) at libzfs_sendrecv.c:2950
#6 0x0000000801077373 in zfs_send_one (zhp=<value optimized out>, from=0x7fffffffeda8 "Ark",
fd=<value optimized out>, flags=<value optimized out>) at libzfs_sendrecv.c:2079
#7 0x000000000040a5c4 in zfs_do_receive (argc=<value optimized out>, argv=0x7fffffffefeb30)
at /freenas-11-releng/freenas/_BE/os/cddl/sbin/zfs/../../../../cddl/contrib/opensolaris/cmd/zfs/zfs_main.c:41
48
#8 0x0000000000000000 in ?? ()
```

and there's

```
assert(size != 0);
```

Due to all the optimizations, I'm not entirely sure how reliable that is. Alexander, do you have any thoughts while I try to look at the code?

#7 - 09/18/2017 12:21 PM - Dru Lavigne

- Status changed from 15 to Unscreened

#8 - 09/18/2017 12:34 PM - Sean Fagan

Another thing to try is to use '-n' with the zfs receive, to see if it dies that way as well.

#9 - 09/18/2017 01:14 PM - Dan Boothby

The transfer successfully ran through the whole snapshot with the -n flag on the receive side.

#10 - 10/03/2017 03:15 AM - Alexander Motin

- Status changed from Unscreened to 15

I have no good ideas, and backtrace from core dump makes no sense to me.

But looking on your debug, I see Ark pool is pretty full and having other datasets. From where did they appear if as you say receive fails after the first dataset? Or you are trying to do incremental replication? In that case your command is just not correct.

#11 - 10/03/2017 05:23 AM - Dan Boothby

As per my original post, it threw the same error when Ark was a brand new empty zpool/dataset. I'm afraid what you're probably seeing on the debug was a attempt down the line (with the same resulting error) after trying to replicate sub-datasets instead of the whole Tank dataset.

I'm also afraid that I now have a dataset on Ark that I need to keep (Ark/NextCloud), so I can't destroy everything if you wanted a clean log.

This is my Tank/Dataset structure as per zfs list:

NAME	USED	AVAIL	REFER	MOUNTPOINT
Tank	9.70T	519G	947K	/mnt/Tank
Tank/Backups	154G	519G	180K	/mnt/Tank/Backups
Tank/Backups/DadsLaptop	58.4G	519G	58.4G	/mnt/Tank/Backups/DadsLaptop
Tank/Backups/LinasLaptop	29.6G	519G	29.6G	/mnt/Tank/Backups/LinasLaptop
Tank/Backups/MediaConfigBackups	314K	519G	314K	/mnt/Tank/Backups/MediaConfigBackups
Tank/Backups/MumsLaptop	66.5G	519G	66.5G	/mnt/Tank/Backups/MumsLaptop
Tank/Media	9.40T	519G	215K	/mnt/Tank/Media
Tank/Media/Downloads	4.17G	519G	3.05G	/mnt/Tank/Media/Downloads
Tank/Media/Files	477K	519G	477K	/mnt/Tank/Media/Files
Tank/Media/Movies	3.95T	519G	3.95T	/mnt/Tank/Media/Movies
Tank/Media/TV Shows	5.45T	519G	5.45T	/mnt/Tank/Media/TV Shows
Tank/Media/Torrents	145K	519G	145K	/mnt/Tank/Media/Torrents
Tank/Vault	149G	519G	149G	/mnt/Tank/Vault

Using zfs send with any recursive snapshot will fail after the first snapshot was sent, so if I tried to send the whole Tank snapshot, or even just the Tank/Backups snapshot, it would for example send the first snapshot of /Tank/Backup/DadsLaptop, transfer the whole thing and then error and not send the next snapshot/dataset (e.g. /Tank/Backup/LinasLaptop).

If I specify the deepest dataset of a recursive snapshot (i.e. only send the snapshot for Tank/Backups/DadsLaptop) it will complete successfully as there is only the one snapshot/dataset to send.

#12 - 10/03/2017 07:14 AM - Alexander Motin

I suppose your pool at some point was used with Coral, at least I see some properties set by it. One of bad side of Coral was its idea to slightly change ZFS in incompatible way. I suspect that your problem is a legacy of that time. Right now I don't see bad things, but could you check that you don't have any snapshots left from that time that may still have incompatible options?

#13 - 10/03/2017 07:16 AM - Alexander Motin

- Related to Bug #24732: The replication failed for the local ZFS volume1/**** while attempting to apply incremental send of snapshot added

#14 - 10/03/2017 08:30 AM - Dan Boothby

The only other snapshots are from Jails, which are on a separate zpool.

I installed corral for all of 15minutes to have a look at the UI, as I was under the impression it wouldn't touch the existing zpools unless I updated the feature flags; was that not the case? :-/

#15 - 10/03/2017 08:44 AM - Alexander Motin

Part of the problem of Coral is that it did not add new feature flags, so there was no real update or compatibility checks.

Just to be sure, please check/show output of `zfs list -t all -r Tank`.

#16 - 10/03/2017 10:00 AM - Dan Boothby

Currently I only have a snapshot of the Media dataset, as I started to try copying each nested dataset separately, but never finished that process (it was working, but I haven't found the time).

When I started the original snapshot copy process I didn't have any snapshots on Tank at all, and have deleted and recreated them as part of trying to troubleshoot this issue a handful of times.

The output is below incase it's showing something the GUI doesn't:

```
Tank
NAME                                USED  AVAIL  REFER  MOUNTPOINT
Tank                                9.72T  500G   947K   /mnt/Tank
Tank/.system                        962M   500G   961M   /mnt/Tank/.system
Tank/.system/configs-5ece5c906a8f4df886779fae5cade8a5 140K   500G   140K   /mnt/Tank/.system/configs-5ece5c906a8f4df886779fae5cade8a5
Tank/.system/cores                  140K   500G   140K   /mnt/Tank/.system/cores
Tank/.system/rrd-5ece5c906a8f4df886779fae5cade8a5 140K   500G   140K   /mnt/Tank/.system/rrd-5ece5c906a8f4df886779fae5cade8a5
Tank/.system/samba4                 186K   500G   186K   /mnt/Tank/.system/samba4
Tank/.system/syslog-5ece5c906a8f4df886779fae5cade8a5 296K   500G   296K   /mnt/Tank/.system/syslog-5ece5c906a8f4df886779fae5cade8a5
Tank/Backups                        154G   500G   180K   /mnt/Tank/Backups
Tank/Backups/DadsLaptop              58.4G  500G   58.4G  /mnt/Tank/Backups/DadsLaptop
Tank/Backups/LinasLaptop             29.6G  500G   29.6G  /mnt/Tank/Backups/LinasLaptop
Tank/Backups/MediaConfigBackups     314K   500G   314K   /mnt/Tank/Backups/MediaConfigBackups
Tank/Backups/MumsLaptop              66.5G  500G   66.5G  /mnt/Tank/Backups/MumsLaptop
Tank/Media                           9.42T  500G   215K   /mnt/Tank/Media
Tank/Media@media-20170925            0      -     215K   -
Tank/Media/Downloads                14.0G  500G   12.9G  /mnt/Tank/Media/Downloads
Tank/Media/Downloads@media-20170925 1.13G  -     2.53G  -
Tank/Media/Files                     477K   500G   477K   /mnt/Tank/Media/Files
Tank/Media/Files@media-20170925      0      -     477K   -
Tank/Media/Movies                   3.96T  500G   3.96T  /mnt/Tank/Media/Movies
Tank/Media/Movies@media-20170925    4.20G  -     3.95T  -
Tank/Media/TV Shows                  5.45T  500G   5.45T  /mnt/Tank/Media/TV Shows
Tank/Media/TV Shows@media-20170925 106M   -     5.44T  -
Tank/Media/Torrents                 145K   500G   145K   /mnt/Tank/Media/Torrents
Tank/Media/Torrents@media-20170925 0      -     145K   -
Tank/Vault                          149G   500G   149G   /mnt/Tank/Vault
root@freenas:~ #
```

#17 - 10/03/2017 10:22 AM - Alexander Motin

I again tried to review properties of your dataset and I see no big criminal. Can you try to create a new dataset on Tank, snapshot it and try to replicate? If it succeed, then I would try to look on some of datasets that fails for the list of modified properties and may be delete/inherit some of them to check whether it help.

#18 - 10/08/2017 05:51 AM - Dan Boothby

I've had chance to create some test datasets, and the recursive snapshot migration has worked for those.

A sample of the zfs properties for a failing dataset vs the new one is below, is that all you needed, or was there something more?

Failing:

```
root@freenas:~ # zfs get all /mnt/Tank/Media/
NAME          PROPERTY          VALUE          SOURCE
Tank/Media    type              filesystem     -
Tank/Media    creation          Tue Dec 15 12:25 2015 -
Tank/Media    used              9.41T         -
Tank/Media    available         500G          -
Tank/Media    referenced        215K          -
Tank/Media    compressratio     1.00x         -
Tank/Media    mounted           yes            -
Tank/Media    quota             none           local
Tank/Media    reservation       none           local
Tank/Media    recordsize        128K          default
Tank/Media    mountpoint        /mnt/Tank/Media default
Tank/Media    sharenfs          off            default
Tank/Media    checksum          on             default
Tank/Media    compression       lz4            local
Tank/Media    atime             off            inherited from Tank
Tank/Media    devices           on             default
Tank/Media    exec              on             default
Tank/Media    setuid            on             default
Tank/Media    readonly          off            default
Tank/Media    jailed            off            default
Tank/Media    snapdir           hidden         default
Tank/Media    aclmode           passthrough   local
Tank/Media    aclinherit        passthrough   inherited from Tank
Tank/Media    canmount          on             default
Tank/Media    xattr             off            temporary
Tank/Media    copies            1             default
Tank/Media    version           5             -
Tank/Media    utf8only          off            -
Tank/Media    normalization     none           -
Tank/Media    casesensitivity    sensitive     -
Tank/Media    vscan             off            default
Tank/Media    nbmand            off            default
Tank/Media    sharesmb          off            default
Tank/Media    refquota          none           local
Tank/Media    refreservation    none           local
Tank/Media    primarycache      all            default
Tank/Media    secondarycache    all            default
Tank/Media    usedbysnapshots   11.6K         -
Tank/Media    usedbydataset     215K          -
Tank/Media    usedbychildren    9.41T         -
Tank/Media    usedbyrefreservation 0             -
Tank/Media    logbias           latency       default
Tank/Media    dedup             off            inherited from Tank
Tank/Media    mlslabel          -             -
Tank/Media    sync              standard      default
Tank/Media    refcompressratio  1.00x         -
Tank/Media    written           11.6K         -
Tank/Media    logicalused       9.42T         -
Tank/Media    logicalreferenced 52.5K         -
Tank/Media    volmode           default       default
Tank/Media    filesystem_limit   none           default
Tank/Media    snapshot_limit     none           default
Tank/Media    filesystem_count   none           default
Tank/Media    snapshot_count     none           default
Tank/Media    redundant_metadata all            default
Tank/Media    org.freenas:description local
```

Working TestData:

```

root@freenas:~ # zfs get all /mnt/Tank/TestData/
NAME                PROPERTY                VALUE                SOURCE
Tank/TestData       type                    filesystem            -
Tank/TestData       creation                Sun Oct  8 13:28 2017 -
Tank/TestData       used                    6.22G                -
Tank/TestData       available               500G                 -
Tank/TestData       referenced              128K                  -
Tank/TestData       compressratio           1.00x                 -
Tank/TestData       mounted                 yes                   -
Tank/TestData       quota                   none                  default
Tank/TestData       reservation             none                  default
Tank/TestData       recordsize              128K                  default
Tank/TestData       mountpoint               /mnt/Tank/TestData   default
Tank/TestData       sharenfs                 off                   default
Tank/TestData       checksum                 on                    default
Tank/TestData       compression              lz4                   inherited from Tank
Tank/TestData       atime                    off                   inherited from Tank
Tank/TestData       devices                  on                    default
Tank/TestData       exec                     on                    default
Tank/TestData       setuid                   on                    default
Tank/TestData       readonly                 off                   default
Tank/TestData       jailed                   off                   default
Tank/TestData       snapdir                  hidden                default
Tank/TestData       aclmode                  passthrough           inherited from Tank
Tank/TestData       aclinherit                passthrough           inherited from Tank
Tank/TestData       canmount                 on                    default
Tank/TestData       xattr                    off                   temporary
Tank/TestData       copies                   1                    default
Tank/TestData       version                  5                    -
Tank/TestData       utf8only                 off                   -
Tank/TestData       normalization            none                  -
Tank/TestData       casesensitivity           sensitive              -
Tank/TestData       vsync                    off                   default
Tank/TestData       nbmand                   off                   default
Tank/TestData       sharesmb                  off                   default
Tank/TestData       refquota                 none                  default
Tank/TestData       refreservation           none                  default
Tank/TestData       primarycache              all                   default
Tank/TestData       secondarycache            all                   default
Tank/TestData       usedbysnapshots           0                    -
Tank/TestData       usedbydataset             128K                  -
Tank/TestData       usedbychildren            6.22G                 -
Tank/TestData       usedbyrefreservation      0                    -
Tank/TestData       logbias                   latency               default
Tank/TestData       dedup                     off                   inherited from Tank
Tank/TestData       mslabel                  -                     -
Tank/TestData       sync                      standard              default
Tank/TestData       refcompressratio          1.00x                 -
Tank/TestData       written                   0                    -
Tank/TestData       logicalused               6.23G                 -
Tank/TestData       logicalreferenced         36.5K                 -
Tank/TestData       volmode                   default               default
Tank/TestData       filesystem_limit          none                  default
Tank/TestData       snapshot_limit            none                  default
Tank/TestData       filesystem_count          none                  default
Tank/TestData       snapshot_count            none                  default
Tank/TestData       redundant_metadata        all                   default
Tank/TestData       org.freenas:description  org.freenas:description local
Tank/TestData       org.freenas:permissions_type PERM                  inherited from Tank

```

#19 - 10/09/2017 02:00 AM - Alexander Motin

Can you try now to inherit different properties with source "local"?

#20 - 10/11/2017 02:49 AM - Alexander Motin

- *Priority changed from Expected to Important*
- *Target version changed from 11.1 to 11.2-BETA1*

I've tried to reproduce it by setting the same properties to dataset on my tests system and it replicated fine. There must be something special with those datasets not visible directly.

#21 - 10/11/2017 05:52 AM - Dan Boothby

Alexander Motin wrote:

Can you try now to inherit different properties with source "local"?

Sorry, I'm not quite that adept at zfs, how would I do this?

#22 - 10/11/2017 06:20 AM - Alexander Motin

```
zfs inherit <property> <dataset>
```

#23 - 10/13/2017 04:56 AM - Dan Boothby

Given that this works on a new dataset with my system, and you've also been able to test this as working, I'm willing to close this ticket as a weird Corral hangover.

As that this was only to be a one time transfer, I'll use rsync for now and hope the new zpool will be error free if I ever need to do this again! Thanks for your help.

#24 - 10/13/2017 08:11 AM - Dru Lavigne

- *Status changed from 15 to Closed: Cannot reproduce*
- *Target version changed from 11.2-BETA1 to N/A*

Thanks for the update Dan. Closing out.

#25 - 10/13/2017 08:12 AM - Dru Lavigne

- *File deleted (debug-freenas-20170918174010.tgz)*

#26 - 10/13/2017 08:12 AM - Dru Lavigne

- *File deleted (zfs.core)*

#27 - 10/13/2017 08:12 AM - Dru Lavigne

- Private changed from Yes to No