

FreeNAS - Bug #27534

Fix iocage traceback when configuration is missing

01/01/2018 06:04 PM - Kevin Horton

Status:	Done		
Priority:	No priority		
Assignee:	Brandon Schneider		
Category:	Middleware		
Target version:	11.1-U2		
Seen in:	11.1	Needs Merging:	No
Severity:	New	Needs Automation:	No
Reason for Closing:	Not to be fixed	Support Suite Ticket:	n/a
Reason for Blocked:		Hardware Configuration:	Bartosz - iocage has done that from the beginning :) The issue is the zfs mountpoints being transferred. Sounds like this is fixed in the next replication incarnation so marking closed.
Needs QA:	Yes	ChangeLog Required:	No
Needs Doc:	No		

Description

The data related to two of my three iocage jails has disappeared from /mnt/iocage, which makes iocage very unhappy. The jails are still seen in the output of jls, and jexec can be used to enter the jails. /mnt/iocage/jails did show all created jails a few hours ago, but two of the three disappeared at some point, without any obvious user action as a trigger.

This may possibly be related to the periodic, recursive snapshots that are made on my main pool, combined with the replication of said snapshots to the backup pool. zfs list shows both main/iocage and backup/iocage as being mounted at /mnt/iocage

```
big_bertha# iocage list
```

```
Traceback (most recent call last):
```

```
File "/usr/local/lib/python3.6/site-packages/iocage/lib/ioc_json.py", line 205, in json_load
    with open(self.location + "/config.json", "r") as conf:
FileNotFoundError: [Errno 2] No such file or directory: '/mnt/iocage/jails/nc/config.json'
```

During handling of the above exception, another exception occurred:

```
Traceback (most recent call last):
```

```
File "/usr/local/bin/iocage", line 10, in <module>
    sys.exit(cli())
File "/usr/local/lib/python3.6/site-packages/click/core.py", line 722, in __call__
    return self.main(*args, **kwargs)
File "/usr/local/lib/python3.6/site-packages/click/core.py", line 697, in main
    rv = self.invoke(ctx)
File "/usr/local/lib/python3.6/site-packages/click/core.py", line 1066, in invoke
    return _process_result(sub_ctx.command.invoke(sub_ctx))
File "/usr/local/lib/python3.6/site-packages/click/core.py", line 895, in invoke
    return ctx.invoke(self.callback, **ctx.params)
File "/usr/local/lib/python3.6/site-packages/click/core.py", line 535, in invoke
    return callback(*args, **kwargs)
File "/usr/local/lib/python3.6/site-packages/iocage/cli/list.py", line 114, in cli
    dataset_type, header, _long, _sort, plugin=plugins, quick=quick)
File "/usr/local/lib/python3.6/site-packages/iocage/lib/iocage.py", line 1194, in list
    exit_on_error=self.exit_on_error).list_datasets()
File "/usr/local/lib/python3.6/site-packages/iocage/lib/ioc_list.py", line 76, in list_datasets
    _all = self.list_all(ds)
File "/usr/local/lib/python3.6/site-packages/iocage/lib/ioc_list.py", line 154, in list_all
```

```
conf = iocage.lib.ioc_json.IOCJson(mountpoint).json_load()
File "/usr/local/lib/python3.6/site-packages/iocage/lib/ioc_json.py", line 302, in json_load
self.json_convert_from_zfs(uuid, skip=skip)
UnboundLocalError: local variable 'uuid' referenced before assignment
```

```
big_bertha# ls -al /mnt/iocage/jails
total 19
drwxr-xr-x  3 root  wheel  3 Jan  1 17:54 .
drwxr-xr-x  8 root  wheel  9 Jan  1 17:53 ..
drwxr-xr-x  3 root  wheel  5 Jan  1 12:05 plex2
```

```
big_bertha# jls
  JID  IP Address      Hostname      Path
  ---  -
   1           owncloud4    /mnt/main/jails/owncloud4
   3 192.168.0.222  handbrake2   /mnt/iocage/jails/handbrake2/root
   4 192.168.0.221  plex2        /mnt/iocage/jails/plex2/root
   5 192.168.0.240  nc           /mnt/iocage/jails/nc/root
```

```
big_bertha# zfs list -o name,mountpoint | grep iocage
backup/iocage                               /mnt/iocage
backup/iocage/download                      /mnt/iocage/download
backup/iocage/download/11.1-RELEASE         /mnt/iocage/download/11.1-RELEASE
backup/iocage/images                        /mnt/iocage/images
backup/iocage/jails                         /mnt/iocage/jails
backup/iocage/jails/handbrake2              /mnt/iocage/jails/handbrake2
backup/iocage/jails/handbrake2/root         /mnt/iocage/jails/handbrake2/root
backup/iocage/jails/nc                     /mnt/iocage/jails/nc
backup/iocage/jails/nc/root                 /mnt/iocage/jails/nc/root
backup/iocage/jails/plex2                   /mnt/iocage/jails/plex2
backup/iocage/jails/plex2/root              /mnt/iocage/jails/plex2/root
backup/iocage/log                           /mnt/iocage/log
backup/iocage/releases                      /mnt/iocage/releases
backup/iocage/releases/11.1-RELEASE         /mnt/iocage/releases/11.1-RELEASE
backup/iocage/releases/11.1-RELEASE/root    /mnt/iocage/releases/11.1-RELEASE/r
oot
backup/iocage/templates                     /mnt/iocage/templates
main/iocage                                 /mnt/iocage
main/iocage/download                        /mnt/iocage/download
main/iocage/download/11.1-RELEASE           /mnt/iocage/download/11.1-RELEASE
main/iocage/images                          /mnt/iocage/images
main/iocage/jails                           /mnt/iocage/jails
main/iocage/jails/handbrake2                /mnt/iocage/jails/handbrake2
main/iocage/jails/handbrake2/root           /mnt/iocage/jails/handbrake2/root
main/iocage/jails/nc                        /mnt/iocage/jails/nc
main/iocage/jails/nc/root                   /mnt/iocage/jails/nc/root
main/iocage/jails/plex2                     /mnt/iocage/jails/plex2
main/iocage/jails/plex2/root                /mnt/iocage/jails/plex2/root
main/iocage/log                             /mnt/iocage/log
main/iocage/releases                        /mnt/iocage/releases
main/iocage/releases/11.1-RELEASE           /mnt/iocage/releases/11.1-RELEASE
main/iocage/releases/11.1-RELEASE/root      /mnt/iocage/releases/11.1-RELEASE/r
oot
main/iocage/templates                       /mnt/iocage/templates
```

History

#1 - 01/02/2018 08:24 AM - Dru Lavigne

- Assignee changed from Release Council to Brandon Schneider
- Target version set to 11.2-BETA1

#2 - 01/04/2018 09:53 PM - Brandon Schneider

- Status changed from Unscreened to Closed: Not To Be Fixed

This is likely because those steps you mention are forcefully unmounting the jails root.

Nothing iocage does will be unmounting the jail. I suggest running `zfs mount -a` and figuring out why/what is causing these datasets to be unmounted.

#3 - 01/05/2018 04:22 AM - Dru Lavigne

- *Target version changed from 11.2-BETA1 to N/A*

#4 - 01/05/2018 06:26 AM - Kevin Horton

Brandon Schneider wrote:

This is likely because those steps you mention are forcefully unmounting the jails root.

Nothing iocage does will be unmounting the jail. I suggest running `zfs mount -a` and figuring out why/what is causing these datasets to be unmounted.

"zfs mount -a" does not cause the missing jails to reappear. They are still missing from `/mnt/iocage/jails`, yet still shown by "jls" and still can be entered via "jexec"

#5 - 01/05/2018 07:30 AM - Dru Lavigne

- *Status changed from Closed: Not To Be Fixed to 46*

- *Target version changed from N/A to 11.1-U1*

#6 - 01/08/2018 05:08 AM - Brandon Schneider

Kevin Horton wrote:

Brandon Schneider wrote:

This is likely because those steps you mention are forcefully unmounting the jails root.

Nothing iocage does will be unmounting the jail. I suggest running `zfs mount -a` and figuring out why/what is causing these datasets to be unmounted.

"zfs mount -a" does not cause the missing jails to reappear. They are still missing from `/mnt/iocage/jails`, yet still shown by "jls" and still can be entered via "jexec"

What does ``mount`` show? I'm suspecting the pool may have done something funky and reimported and they are scattered. If they show in jls, you can always jail -r JID of the jail. That should allow you to troubleshoot without worrying about the processes in the jail running.

#7 - 01/08/2018 05:11 AM - Dru Lavigne

- Status changed from 46 to 15

#8 - 01/08/2018 06:46 AM - Kevin Horton

Brandon Schneider wrote:

Kevin Horton wrote:

Brandon Schneider wrote:

This is likely because those steps you mention are forcefully unmounting the jails root.

Nothing iocage does will be unmounting the jail. I suggest running `zfs mount -a` and figuring out why/what is causing these datasets to be unmounted.

"`zfs mount -a`" does not cause the missing jails to reappear. They are still missing from `/mnt/iocage/jails`, yet still shown by "jls" and still can be entered via "jexec"

What does ``mount`` show? I'm suspecting the pool may have done something funky and reimported and they are scattered. If they show in jls, you can always `jail -r JID of the jail`. That should allow you to troubleshoot without worrying about the processes in the jail running.

I restarted the system yesterday, as I had suddenly lost the ability to enter one of the jails using "jexec". I had rebooted twice before in an attempt to clear the issue, but those earlier reboots had not resolved it. This latest reboot seems to have resolved the problem, at least for the moment. The iocage jails have reappeared, and iocage is working normally.

"mount" shows the jails on the main pool mounted at `/mnt/iocage`, and the ones replicated to the backup pool are mounted at `/mnt/backup/iocage`. Earlier, when the problem was evident, the jails on the main and backup pools were both mounted at `/mnt/iocage`.

Note: I am aware of another user with the same or a similar problem. He may be able to provide information from a system that currently exhibits the issue.

#9 - 01/09/2018 06:45 AM - Dru Lavigne

- Status changed from 15 to Closed: Not Applicable
- Target version changed from 11.1-U1 to N/A

Kevin: I'll close this out for now. If you or the other user can recreate, please attach a debug (System -> Advanced -> Save Debug) from the system to this ticket.

#10 - 01/10/2018 12:57 AM - Heiko Kirschke

- File debug-heiko-fs-20180110094659.tgz added

Here's my debug log. My system shows the behaviour as explained by Kevin in his first message, the correct /mnt/iocage mounts from qandd/iocage get shadowed by the ones of backup/qandd/iocage. This is the situation directly after a reboot, without having called iocage. I'm also using periodic snapshots and a replication task to back up from local disk `qandd` to local disk (resp. dataset) `backup/qandd`. I'll leave my system in that state, so come back to me if you need further analysis. (My 2 cents: Without knowing the details, I guess a search pattern within iocage greps too many mount points.)

In the meantime I found a workaround: When calling

```
zfs umount -f backup/qandd/iocage
```

I get back the initial mounts for /mnt/iocage, and everything works fine. I have to repeat that workaround every hour at :00, since some process does that mount from backup/qandd/iocage to /mnt/iocage again. When I disable the replication task for the dataset carrying the iocage dataset to the backup volume, the shadowing mounts are not done any longer.

```
heiko-fs# mount|grep iocage
qandd/iocage on /mnt/iocage (zfs, local, nfsv4acls)
qandd/iocage/download on /mnt/iocage/download (zfs, local, nfsv4acls)
qandd/iocage/download/11.1-RELEASE on /mnt/iocage/download/11.1-RELEASE (zfs, local, nfsv4acls)
qandd/iocage/images on /mnt/iocage/images (zfs, local, nfsv4acls)
qandd/iocage/jails on /mnt/iocage/jails (zfs, local, nfsv4acls)
qandd/iocage/jails/vdr11 on /mnt/iocage/jails/vdr11 (zfs, local, nfsv4acls)
qandd/iocage/jails/vdr11/root on /mnt/iocage/jails/vdr11/root (zfs, local, nfsv4acls)
qandd/iocage/log on /mnt/iocage/log (zfs, local, nfsv4acls)
qandd/iocage/releases on /mnt/iocage/releases (zfs, local, nfsv4acls)
qandd/iocage/releases/11.1-RELEASE on /mnt/iocage/releases/11.1-RELEASE (zfs, local, nfsv4acls)
qandd/iocage/releases/11.1-RELEASE/root on /mnt/iocage/releases/11.1-RELEASE/root (zfs, local, nfsv4acls)
qandd/iocage/templates on /mnt/iocage/templates (zfs, local, nfsv4acls)
backup/qandd/iocage on /mnt/iocage (zfs, local, read-only, nfsv4acls)
backup/qandd/iocage/download on /mnt/iocage/download (zfs, local, read-only, nfsv4acls)
backup/qandd/iocage/download/11.1-RELEASE on /mnt/iocage/download/11.1-RELEASE (zfs, local, read-only, nfsv4acls)
backup/qandd/iocage/images on /mnt/iocage/images (zfs, local, read-only, nfsv4acls)
backup/qandd/iocage/jails on /mnt/iocage/jails (zfs, local, read-only, nfsv4acls)
backup/qandd/iocage/jails/vdr11 on /mnt/iocage/jails/vdr11 (zfs, local, read-only, nfsv4acls)
backup/qandd/iocage/jails/vdr11/root on /mnt/iocage/jails/vdr11/root (zfs, local, read-only, nfsv4acls)
backup/qandd/iocage/log on /mnt/iocage/log (zfs, local, read-only, nfsv4acls)
backup/qandd/iocage/releases on /mnt/iocage/releases (zfs, local, read-only, nfsv4acls)
backup/qandd/iocage/releases/11.1-RELEASE on /mnt/iocage/releases/11.1-RELEASE (zfs, local, read-only, nfsv4acls)
backup/qandd/iocage/releases/11.1-RELEASE/root on /mnt/iocage/releases/11.1-RELEASE/root (zfs, local, read-only, nfsv4acls)
backup/qandd/iocage/templates on /mnt/iocage/templates (zfs, local, read-only, nfsv4acls)
```

#11 - 01/10/2018 04:25 AM - Dru Lavigne

- Status changed from Closed: Not Applicable to Unscreened
- Target version changed from N/A to 11.1-U2

#12 - 01/12/2018 09:50 AM - Brandon Schneider

- Status changed from Unscreened to Screened

Screening this as the fix in iocage will be to avoid the traceback.

For those wondering how this is happening, iocage list actually parses jls, so the jails are "technically" running and available. But the mountpoint is shadowed, so every operation will fail. Considering iocage doesn't actually mount its dataset, but relies on it to be already mounted, this behavior isn't caused by iocage, as Heiko also describes seeing it without issuing iocage at all. But like Heiko also says, there is some other process that's causing these to shadow mount each other.

This will be assigned to Bartosz afterwards to investigate if it's the replication at fault.

#13 - 01/15/2018 03:44 PM - Ben Gadd

- Due date set to 02/02/2018

#14 - 01/15/2018 07:21 PM - Kevin Horton

One observation that may or may not be related: when I created my first jail using iocage, the system did not require a user input to specify which pool to use for the iocage jails. The system put the iocage jails on my backup pool, which was the first pool when sorted alphabetically. I desired that the jails be on the "main" pool, as that was the pool that was replicated to my second NAS. I then used "iocage activate main" to specify that the iocage jails should be on the "main" pool. I suspect that this resulted in both /mnt/backup/iocage and /mnt/main/iocage being mounted at /mnt/iocage.

#15 - 01/16/2018 12:29 AM - Heiko Kirschke

That's exactly what I did also. The root cause for this problem is that "iocage activate" sets the zfs property "mountpoint" to "/mnt/iocage". That zfs mountpoint property is replicated from the iocage activated dataset (qandd) to the backup dataset, both pointing to /mnt/iocage, see the output of zfs get below.

```
heiko-fs# zfs get -r all qandd/iocage|grep mnt
qandd/iocage                                mountpoint                                /mnt/iocage
qandd/iocage/download                       local
qandd/iocage/download/11.1-RELEASE         mountpoint                                /mnt/iocage/download
qandd/iocage/download/11.1-RELEASE/11.1-RELEASE inherited from qandd/iocage
qandd/iocage/images                         mountpoint                                /mnt/iocage/images
qandd/iocage/jails                          inherited from qandd/iocage
qandd/iocage/jails/vdr11                    mountpoint                                /mnt/iocage/jails
qandd/iocage/jails/vdr11/11.1-RELEASE      inherited from qandd/iocage
qandd/iocage/jails/vdr11/11.1-RELEASE/root mountpoint                                /mnt/iocage/jails
qandd/iocage/log                            inherited from qandd/iocage
qandd/iocage/releases                       mountpoint                                /mnt/iocage/releases
qandd/iocage/releases/11.1-RELEASE         mountpoint                                /mnt/iocage/releases
qandd/iocage/releases/11.1-RELEASE/11.1-RELEASE inherited from qandd/iocage
qandd/iocage/releases/11.1-RELEASE/11.1-RELEASE/root mountpoint                                /mnt/iocage/releases
qandd/iocage/templates                     mountpoint                                /mnt/iocage/templates
qandd/iocage/templates/11.1-RELEASE       inherited from qandd/iocage
```

```
heiko-fs# zfs get -r all backup/qandd|grep mnt
```

backup/qandd		mountpoint	/
mnt/backup/qandd	default		
backup/qandd/iocage		mountpoint	/
mnt/iocage	received		
backup/qandd/iocage/download		mountpoint	/
mnt/iocage/download	inherited from backup/qandd/iocage		
backup/qandd/iocage/download/11.1-RELEASE		mountpoint	/
mnt/iocage/download/11.1-RELEASE	inherited from backup/qandd/iocage		
backup/qandd/iocage/images		mountpoint	/
mnt/iocage/images	inherited from backup/qandd/iocage		
backup/qandd/iocage/jails		mountpoint	/
mnt/iocage/jails	inherited from backup/qandd/iocage		
backup/qandd/iocage/jails/vdr11		mountpoint	/
mnt/iocage/jails/vdr11	inherited from backup/qandd/iocage		
backup/qandd/iocage/jails/vdr11/root		mountpoint	/
mnt/iocage/jails/vdr11/root	inherited from backup/qandd/iocage		
backup/qandd/iocage/log		mountpoint	/
mnt/iocage/log	inherited from backup/qandd/iocage		
backup/qandd/iocage/releases		mountpoint	/
mnt/iocage/releases	inherited from backup/qandd/iocage		
backup/qandd/iocage/releases/11.1-RELEASE		mountpoint	/
mnt/iocage/releases/11.1-RELEASE	inherited from backup/qandd/iocage		
backup/qandd/iocage/releases/11.1-RELEASE/root		mountpoint	/
mnt/iocage/releases/11.1-RELEASE/root	inherited from backup/qandd/iocage		
backup/qandd/iocage/templates		mountpoint	/
mnt/iocage/templates	inherited from backup/qandd/iocage		

Workaround is to set the mountpoint property of the receiving backup dataset back to "inherit":

```
heiko-fs# zfs inherit mountpoint backup/qandd/iocage
heiko-fs# zfs get -r all backup/qandd|grep mnt
```

backup/qandd		mountpoint	/
mnt/backup/qandd	default		
backup/qandd/iocage		mountpoint	/
mnt/backup/qandd/iocage	default		
backup/qandd/iocage/download		mountpoint	/
mnt/backup/qandd/iocage/download	default		
backup/qandd/iocage/download/11.1-RELEASE		mountpoint	/
mnt/backup/qandd/iocage/download/11.1-RELEASE	default		
backup/qandd/iocage/images		mountpoint	/
mnt/backup/qandd/iocage/images	default		
backup/qandd/iocage/jails		mountpoint	/
mnt/backup/qandd/iocage/jails	default		
backup/qandd/iocage/jails/vdr11		mountpoint	/
mnt/backup/qandd/iocage/jails/vdr11	default		
backup/qandd/iocage/jails/vdr11/root		mountpoint	/
mnt/backup/qandd/iocage/jails/vdr11/root	default		
backup/qandd/iocage/log		mountpoint	/
mnt/backup/qandd/iocage/log	default		
backup/qandd/iocage/releases		mountpoint	/
mnt/backup/qandd/iocage/releases	default		
backup/qandd/iocage/releases/11.1-RELEASE		mountpoint	/
mnt/backup/qandd/iocage/releases/11.1-RELEASE	default		
backup/qandd/iocage/releases/11.1-RELEASE/root		mountpoint	/
mnt/backup/qandd/iocage/releases/11.1-RELEASE/root	default		
backup/qandd/iocage/templates		mountpoint	/
mnt/backup/qandd/iocage/templates	default		
backup/qandd/jails		mountpoint	/
mnt/backup/qandd/jails	default		
backup/qandd/jails/ampache		mountpoint	/
mnt/backup/qandd/jails/ampache	default		
backup/qandd/jails/owncloud		mountpoint	/
mnt/backup/qandd/jails/owncloud	default		
backup/qandd/jails/saned		mountpoint	/
mnt/backup/qandd/jails/saned	default		
backup/qandd/jails/transmission_1		mountpoint	/
mnt/backup/qandd/jails/transmission_1	default		
backup/qandd/jails/vdr		mountpoint	/
mnt/backup/qandd/jails/vdr	default		
backup/qandd/qandd		mountpoint	/
mnt/backup/qandd/qandd	default		

To eventually solve this bug, the zfs mountpoint property should be excluded from replication to a local backup volume. This is currently not yet possible using the FreeNAS GUI (old version), since the "zfs recv" process has to be told to replace that property by option "-o", see e.g. https://docs.oracle.com/cd/E23824_01/html/821-1448/gbchx.html#gjonv how to do this. Not sure if this can be handled by FreeNAS, since I saw no field in the GUI where I can provide additional "zfs recv" options.

#16 - 01/24/2018 05:45 AM - Dru Lavigne

- Status changed from Screened to Not Started

#17 - 01/26/2018 06:18 AM - Ben Gadd

- Due date changed from 02/02/2018 to 02/12/2018

Due date updated to reflect the code freeze for 11.1U2.

#18 - 01/26/2018 09:38 AM - Ben Gadd

- Severity set to New

#19 - 02/05/2018 08:53 AM - Brandon Schneider

- Status changed from Not Started to In Progress

#20 - 02/08/2018 06:36 PM - Brandon Schneider

- Status changed from In Progress to Not Started

- Assignee changed from Brandon Schneider to Bartosz Prokop

The traceback should be solved now in: <https://github.com/iocage/iocage/commit/2c32c7f41fd99606f13e57227cfec22342b55da3>

As far as iocage and mounting is concerned, the first activated pool is given the mount '/iocage'. Any additional pool is given the mount '/pool/iocage' to avoid conflicts like this. The duplicate checking behavior was too naive, and has also been fixed in <https://github.com/iocage/iocage/commit/515541180c23faa4b8b45de8ebe058bcafb320f2>.

Passing to Bartosz to make sure mountpoint conflicts on the replicated side are dealt with.

#21 - 02/09/2018 10:49 AM - Bartosz Prokop

- Status changed from Not Started to In Progress

#22 - 02/09/2018 05:56 PM - Bartosz Prokop

- Status changed from In Progress to Not Started

Brandon,

From the legacy/current replication point of view mentioned situation occurs when 'Delete stale snapshots on remote system'(follow_delete in code) is checked. ZFS properties are sent only when this option is enabled(I'm not fully aware of the intentions).

Unfortunately It's too late to play with a fragile autorepl code but I can address those issues in the new replication engine.

I'm using **custom** ZFS properties in the new replication code extensively to store metadata and I'm encouraging you to do the same with IOcage if it is possible.

Please contact me directly if I can do anything before 11.1-u2 to solve this problem.

Bartosz

#23 - 02/09/2018 05:57 PM - Bartosz Prokop

- Assignee changed from Bartosz Prokop to Brandon Schneider

#24 - 02/12/2018 01:58 AM - Brandon Schneider

- Status changed from Not Started to Closed

- Reason for Closing set to Not to be fixed

- Hardware Configuration updated

#25 - 02/12/2018 03:23 AM - Dru Lavigne

- Subject changed from iocage jails disappearing from /mnt/iocage/jails to Fix iocage traceback when configuration is missing

- Status changed from Closed to Done

- Needs Doc changed from Yes to No

#26 - 02/12/2018 03:23 AM - Dru Lavigne

- File deleted (debug-heiko-fs-20180110094659.tgz)

#27 - 02/12/2018 08:37 AM - Dru Lavigne

- Needs Merging changed from Yes to No