Subject: Why the ploop become the default?
Posted by yuri on Sun, 27 Apr 2014 15:15:46 GMT
View Forum Message <> Reply to Message

Hello, this is my first topic here.

I'm starting this discussion because in vzctl 4.7 update, the ploop has been set as default.

I'm against this because there's a lot of problems with ploop solution that are even informed in openvz wiki and a lot of people still don't know.

I will start with the main disadvantages that aren't even cited in OpenVZ Wiki

Ploop / Why Not:

The first of all, the questionable beneficts cited in Wiki:

"File system journal is not bottleneck anymore" - This problem is very easy to solve without use of ploop. First, if your file system isn't the ext4 you don't have this problem. Second, to solve this in ext4 just add "journal_async_commit" to mount options, and the jornal won't lock the disk anymore.

"Number of inodes doesn't have to be limited because this is not a shared resource anymore (each CT has its own file system)" - This is a question depends how you have prepared your FS. Again, if you use a different filesystem than EXT4 you don't have this problem. All other FSs (ReiserFS, JFS, XFS, ZFS and BTRFS) don't limit the inode allocation, and even if you choice for EXT4 you can greatly increase it by diminishing the "bytes per inode" during mkfs.

And now the main problems of Ploop against SIMFS:

- One big file is simplest to work but is easier to crack:
Is much easier to work with just on file, to transfer and backup it. But is easier to crack it too. As happen in others one file disks like VMWare, OpenVZ and others, if you have problems with the ploop file (write erros, transfer problems and any other thing that can crack the ploop file structure) you will probably can't mount it anymore.

- Multi-partition support and multi resize problems:
If you make multi partitions in same ploop, you will have a lot of problems to resize them. Because you cannot increase the size of the non last partition without move the partitions after it. You can try to move the partition to the end to avoid move the others, but this still need to off-line the container and will take o a much more time depending the current used space of it and you will have a big space waste in the ploop.
You can try to use a ploop per partition to avoid these problems. But it will be a bit harder to control the total space and you cannot allow the container owner to manage their own partitions.
If you use a different partition type then EXT4 you cannot do a online resize via OpenVZ tools. If the partition that you choice allow online resize, you have to do it by yourself.

- Ploop space waste and fragmentation:
The ploop have advantages against the old loopbacks files. You don't need to allocate their full max size. But it still have a lot of espace wasted by fragmentation.
The ploop simply cannot see which blocks are in use or are marked as free, so the fragmented delete blocks will occupy the unused space as if they are in use. At last until will some defrag function be implemented in ploop.

And finally, the ploop started deprecated:
Currently the ploop only works on EXT4 and NFS. But the EXT4 are in their end of life and the NFS don't meets the demands of a lot of people for distributed allocation. The most distributions and sysadms are adopting others better and newer FSs, like XFS, BTRFS or ZFS in their place. The most important example is that the RHEL 7 (and their derivates) will adopt the XFS as default FS instead of EXT4.

I know that in future most of these problems will be solved in ploop. But I don't belive that is time to change it to default and keep simfs as secondary option.

I'm sugesting to back the SIMFS for default in next vzctl versions.

The SIMFS is lighter, easier to keep and well tested solution the keep until now and is a wrong decision to change the focus to ploop now. Instead I believe is a better solution to OpenVZ team to try to focus the SIMFS on next generation FS, like BTRFS or even ZFS that can solve the most problems of space allocation and reservation, snapshots support, unlimited inodes instead of make a lot of rework and workarounds in the ploop solution.

Subject: Re: Why the ploop become the default?
Posted by Ales on Tue, 29 Apr 2014 10:56:43 GMT
View Forum Message <> Reply to Message

I think that switching simfs/ploop defaults for existing systems is something that should have been announced well in advance.

Something like "ploop will be the default form the next stable release of vzctl forward, prepare for it" should have been announced... Perhaps I missed it, but I didn't see anything on the web page, user mailing list or in the forum.

As for the first post in this thread, I see some valid points.

Two things I'm in disagreement with:
- EXT4 is not at it's end of life or in any way deprecated. The fact that RHEL 7 is using XFS as default doesn't mean any such thing
- openvz isn't switching it's focus to ploop now... it seems to me that this has happened a good while ago. IMHO, this change in vzctl default settings is just the culmination of the development effort, meant to gather more ploop users quickly.

That being said, I would love if openvz could support simfs on more than just EXT4. Having a valid choice between ZFS, BtrFS, XFS or EXT4 would be great. But I gess it's just too much work.

Subject: Re: Why the ploop become the default?
Posted by yuri on Wed, 30 Apr 2014 14:47:26 GMT
View Forum Message <> Reply to Message

Sorry, but i'm still desagree with the points thay you have appointed Ales:

Quote:- EXT4 is not at it's end of life or in any way deprecated. The fact that RHEL 7 is using XFS as default doesn't mean any such thing
When I said that the EXT4 are deprecated is because it's a 20 years old filesystem, very stable and useful, but full of restrictions and limitations of the old main structure of the EXT2 that is kept for retro compatibility that can't be changed to prevent it to stop working on very old systems.

Things like multi-cpu support for it are almost inexistent, their journal is outsite the main meta-data structure (is a special system hidden file), their index support is a strange system that only work with inodes (can't index dentrys or organize they in a b-tree) when there are more than 100 in same dir and a lot of other things that can't be implemented or implemented in right way.

That the why the most distribution are changing or considering to change their default FS to XFS or BTRFS and placing the EXT4 as an optional one. So I know that are important to keep the support for it, but I don't belive that is better is focus on it like was happen in ploop.

Quote:- openvz isn't switching it's focus to ploop now... it seems to me that this has happened a good while ago. IMHO, this change in vzcWhen I said that the EXT4 are deprecated is becouse it's is a 20 years old filesystem, very stable and useful, but full of restrictions and limitations of the old main structure of the EXT2 retro compatibility that can't be chaged to prevent it to stop working on very old systems.

Things like multi-cpu support for it are almost inexistent, their journal is outsite the main meta-data structure (is a special system hidden file), their index support is a strange system that only work with inodes (can't index dentrys or organize they in a b-tree) when there are more than 100 in same dir and a lot of other things that can't be implemented or implemented in right way.

That the why the most distribution are changing or considering to change their default FS to XFS or BTRFS and placing the EXT4 as an optional one. So I know that are important to keep the support for it, but I don't belive that is better is focus on it like was happen in ploop.tl default settings is just the culmination of the development effort, meant to gather more ploop users quickly.

Sorry but I still belive when you change something to default is because you want that more people use it so you can make more focus on it.

Quote:That being said, I would love if openvz could support simfs on more than just EXT4. Having a valid choice between ZFS, BtrFS, XFS or EXT4 would be great. But I gess it's just too much work.

The SIMFS already support all other FS. The only thing that misses is a vzctl-core support for BTRFS advanced features like snapshots, send/recive, qgroups.... But I know that is early to work on this because the RHEL 7 not even is released yet. This is just an idea for the near future as alternative to use the ploop features with simfs. (like already happen with LVM (but the LVM solution is terrible!!!)).

The main fact that I'm saying is: change the ploop to the default was a wrong decision, it's still lacks support for new FSs both for hardnode and for the VM and these supports can solve most of their problems.

Subject: Re: Why the ploop become the default?
Posted by Ales on Thu, 01 May 2014 02:23:53 GMT
View Forum Message <> Reply to Message

yuri wrote on Wed, 30 April 2014 16:47The SIMFS already support all other FS.

My definitions of the words deprecated and supported are a bit different than yours...

Sure, simfs can work on other file systems as of now, but I think having user quotas in CTs is a prerequisite for labeling file systems as supported. Quotas work only on ext2/3/4 and that alone makes anything but ext4 unusable for many openvz use cases.

Don't get me wrong, having working quotas and BtrFS or ZFS specific functionality would be great. I just don't think openvz project has enough resources to accomplish all this.

Subject: Re: Why the ploop become the default?
Posted by bjdea1 on Thu, 28 Aug 2014 08:12:44 GMT
View Forum Message <> Reply to Message

Ales wrote on Thu, 01 May 2014 12:23yuri wrote on Wed, 30 April 2014 16:47The SIMFS already support all other FS.

My definitions of the words deprecated and supported are a bit different than yours...

Sure, simfs can work on other file systems as of now, but I think having user quotas in CTs is a prerequisite for labeling file systems as supported. Quotas work only on ext2/3/4 and that alone makes anything but ext4 unusable for many openvz use cases.

Don't get me wrong, having working quotas and BtrFS or ZFS specific functionality would be great. I just don't think openvz project has enough resources to accomplish all this.

I use "ZFS on Linux" (ZOL) with simfs and with quotas, on production servers. The trick is to create ZVOLS and then create ext4 filesystems on top of these. That gives you the benefits of an underlying "Copy On write" filesystem (ZOL) and the ext4 quotas on top, making everything work

together.

I use ZOL for compression and stability, its just a much more precise Fileysystem than EXT4 alone. I also have the option of adding an SSD cache any time I need.

So far ZFS on linux Zvols with ext4 on top have been running smoothly for over 6 months.

The only issues I face are a slightly lower performance, but not by much, and updating ZOL is somewhat troublesome. ZOL is in the form of kernel modules which need to be compiled for each kernel update.

I want to upgrade everything to ploop but as pointed out, its not perfect by any means yet and I am still hesitant.

Subject: Re: Why the ploop become the default?
Posted by mangust on Sun, 03 Sep 2017 02:48:19 GMT
View Forum Message <> Reply to Message

Completely agree.

SIMFS is a huge advantage over all this emulated files. They are easy to break, difficult to backup, much more chances to get into some inconsistent state.
SIMFS is transparent, lightweight, reliable. One of advantages I used as an argument to convert many people into ovz religion.
