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.
