From: **Arrigo Triulzi** arrigo@alchemistowl.org
Subject: [Cabal] A Greybeard vs. APFS (or "what happens when wheels are reinvented")
Date: 6 February 2018 at 13:07
To: cabal@alchemistowl.org

AT

I was going to make this a tweetstorm but then realised that perhaps a smaller audience might actually appreciate it.

Basic assumptions:

* filesystems ideally should be designed not to knowingly lose data or behave in a way which encourages data loss.
* APFS is Apple's new filesystem, initially rolled-out on iOS and now on OS X (macOS)
* everyone knows that you don't touch most operating systems until they hit the .3 milestone, especially Apple's

Background:

I've lost data with pretty much every filesystem since the late 70s with two exceptions: ZFS and Tandem's NSK. The worst which ever happened to me was on a flight between London and Milan on a little Jumbolino when I accidentally hit the power button on my Compaq Armada M300 running Linux 2.4 and ReiserFS. When I powered it back up ReiserFS did its recovery and every single file which was open when I hit the power switch was truncated to zero length. As any non-novice Unix user knows the open files on a modern Unix systems include everything from the kernel modules to dynamic libraries plus, of course, user data. I landed in a rather vile mood.

Looking at APFS I recognised one characteristic from a filesystem I knew well: AdvFS-style volume management[1]. AdvFS (_Adv_anced FileSystem[2]) was designed by Digital for their OSF/1 Unix running on Alphas and released in the mid-90s. Nobody really used it until OSF/1 hit version 3.2D, officially called Digital Unix at the time. What AdvFS brought which was of great interest was that you could split a container into volumes which all shared the same space and therefore, unlike UFS partitioning, you no longer had to figure out how to resize /var/spool/mail when the darlings subscribed to linux-users@vger.

On my OS X machines I let Apple do its partitioning (i.e. one huge partition with HFS+ and FileVault 2) and then use encrypted sparse images for my important data which I mount by hand using a trivial script calling hdiutil and typing in the relevant passwords. As you can select mount points I do this, for example, for Library/Mail so that my e-mail is encrypted at rest. I also use an encrypted container for each client project, etc. etc. As this happens on a 30TB external RAID as you can imagine the fsck_hfs runs are painful (they take approximately 12 hrs). The only peculiarity is that there is an admin account on the internal disk and my user account lives on the external RAID volume: as I use FileVault 2 on the RAID array too I have to log in as the admin account so that it mounts the external disks before I can log in as myself. This is totally stupid but Apple in its immense wisdom decided that external FileVault disks cannot be auto-mounted at boot even if you have to type in the password (and yes, I have confirmed this with Apple: "design decision").

As per my standards I resisted High Sierra until 10.13.3 at which point I decided that APFS alone was worth the plunge. I let the installer do its magic and then, on my laptop, I stupidly converted the external Thunderbolt backup disk to APFS thereby losing all my backups because TimeMachine does not allow you to backup on APFS but only on HFS+ (APFS does not have directory hard links which TimeMachine uses). Fortunately this was just my laptop, my desktop was going to be next and I would not make the same mistake again. Off goes the MacPro, the internal 1TB SSD converts happily, then I convert by hand only the user partition of the external RAID leaving the backup partition on HFS+.

APFS vs. Greybeard:

I now log in as the admin user to mount the external filesystem and… nothing mounts from the external RAID. OK, so DiskUtility, mount by hand and the reason was that it had "lost" the passwords, or rather, the passwords stored in KeyChain were no longer recognised as the disk identifiers had changed. Fine, add them back in and the filesystems mount. Log out from the admin user, attempt to log in with my normal user and… hang. Wait for a good 5 minutes and then go for a reset.

1st discovery: OS X now unmounts external disks when you log out. It didn't used to do this.

I therefore need to keep the admin user logged in and switch user to be able to access my home directory.

As usual after an upgrade TimeMachine starts kicking into electrifying action and then dies saying it cannot access the backup partition. Right. What's wrong this time? Nothing apparently as I can read all the files on it and it takes a while (and a Terminal) to discover that the "Administrators" Unix group has been changed from "system" to "admin". A recursive chgrp on the backup disk taking the best part of 30 minutes because of the immense structures in the 8TB HFS+ partition (remember? No conversion or no backup) fixes that issue.

2nd discovery: OS X changes groups and forgets to fix the external disks (the internal snapshots work fine, of course).

Having negotiated this nightmare I now decide to test APFS to make use of its newly found skills, i.e. the pretty volumes in the same container thing. What does a greybeard do? Well, apply the golden rule of using the official tool to do whatever you want to do and therefore open DiskUtility. Create an encrypted volume with an adequate minsize and maxsize (same as good ol' AdvFS). Obviously DiskUtility mounts it as /Volume/untitled which is fine for the time being. <enter moment of joy thinking that I can get rid of my encrypted sparse images>. I therefore decide that I want to now use hdiutil to mount the volume attached to the directory I want to be on: click on the unmount button, get asked if I want to unmount all the volumes on the container (of course not, no) and then move to Terminal. All hung.

on the container (of course not, no) and then move to Terminal. All hung.

3rd discovery: when you unmount a volume within a container, even if you say you just want to unmount that volume, DiskUtility unmounts the whole flipping container.

This is beyond unbelievable because, of course, OS X normally refuses to unmount filesystems which are “in use” like pretty much all Unix operating systems on the planet. But no. To make matters worse what happens is that clearly the user is hung because my personal user is on the external volume which is in the same container as the encrypted volume I had created and had just unmounted. There is no fix except a hard reboot.

4th discovery: APFS is unable to cleanly recover my volume on the external disk.

As the admin user I need to open DiskUtility, run “first aid” for over 20 minutes before the APFS volume manages to mount. The errors only indicate some timestamp inconsistencies but memories of ReiserFS are still with me…

What does a greybeard do? Well, the first assumption is “I’m an idiot, I must have pressed unmount all volumes by accident and brought it upon myself”. I therefore *do it again* because I think that Apple could not possibly be this ridiculously stupid.

5th discovery: my memory is not as bad as I think it is.

This time I decide to do it via hdiutil because it is on the command line and I’m old. What a brilliant idea: for starters the FileVault password prompt is no longer in text on the terminal but pops up a window which, amongst other things, takes away the focus from the Terminal you are into which is bad UI behaviour to begin with. I then successfully mount /dev/disk3s3 and then I decide to unmount it. As I used hdiutil I didn’t even get the prompt as to whether I was interested in blowing up my own disk again. It just did.

Conclusion:

I have no idea of who signed off APFS for production use but he/she should be taken out the back and shot. The reliability and behaviour of the system when a user tries to use the filesystem precisely in the way it was allegedly designed (i.e. create different volumes within the same container) cannot possibly have been tested. Not only, the “new” behaviours which were introduced with respect to external disks are undocumented and totally new to staff at Apple’s “Genius” bar.

The accessory conclusion is that clearly the new mantra with OS X is not only “wait until .3” but stay a full release behind (10.12 was substantially more stable[3]).

We are here discussing security when the most valuable hardware company on the planet is allowed to ship an operating system which doesn’t even pass the most basic reliability requirements (and yes, Windows is no better but that is hardly consolation).

[1] yes, I know ZFS pools do the same, if anything a lot better and precisely how things should be done but I digress… they also happened after AdvFS (at least the release, development was probably almost concurrent).
[2] AdvFS is now free, not that anyone cares “because BtrFS/ext4/other abominations” (http://advfs.sourceforge.net)
[3] cf. my Twitter rants about Terminal in 10.13 crashing with all the open tabs when an IPv6 connection initiated as a “remote connection” within a Terminal tab is terminated abruptly.