



# Aaron Toponce

{ 2013.05.09 }

## ZFS Administration, Appendix B- Using USB Drives

### Table of Contents

#### Zpool Administration

0. [Install ZFS on Debian GNU/Linux](#)
1. [VDEVs](#)
2. [RAIDZ](#)
3. [The ZFS Intent Log \(ZIL\)](#)
4. [The Adjustable Replacement Cache \(ARC\)](#)
5. [Exporting and Importing Storage Pools](#)
6. [Scrub and Resilver](#)
7. [Getting and Setting Properties](#)
8. [Best Practices and Caveats](#)

#### ZFS Administration

9. [Copy-on-write](#)
10. [Creating Filesystems](#)
11. [Compression and Deduplication](#)
12. [Snapshots and Clones](#)
13. [Sending and Receiving Filesystems](#)
14. [ZVOLS](#)
15. [iSCSI, NFS and Samba](#)
16. [Getting and Setting Properties](#)
17. [Best Practices and Caveats](#)

#### Appendices

- A. [Visualizing The ZFS Intent Log \(ZIL\)](#)
- B. [Using USB Drives](#)
- C. [Why You Should Use ECC RAM](#)
- D. [The True Cost Of Deduplication](#)

### Introduction

This comes from the "why didn't I think of this before?!" department. I have lying around my home and office a ton of USB 2.0 thumb drives. I have six 16GB drives and eight 8GB drives. So, 14 drives in total. I have two hypervisors in a GlusterFS storage cluster, and I just happen to have two USB squids, that support 7 USB drives each. Perfect! So, why not put these to good use, and add them as L2ARC devices to my pool?

## Disclaimer

USB 2.0 is limited to 40 MBps per controller. A standard 7200 RPM hard drive can do 100 MBps. So, adding USB 2.0 drives to your pool as a cache is not going to increase the read bandwidth. At least not for large sequential reads. However, the seek latency of a NAND flash device is typically around 1 milliseconds to 3 milliseconds, whereas a platter HDD is around 12 milliseconds. If you do a lot of small random IO, like I do, then your USB drives will actually provide an overall performance increase that HDDs cannot provide.

Also, because there are no moving parts with NAND flash, this is less data that needs to be read from the HDD, which means less movement of the actuator arm, which means consuming less power in the long term. So, not only are they better for small random IO, they're saving you power at the same time! Yay for going green!

Lastly, the L2ARC *should* be read intensive. However, it can also be write intensive if you don't have enough room in your ARC and L2ARC to store all the requested data. If this is the case, you'll be constantly writing to your L2ARC. For USB drives without wear leveling algorithms, you'll chew through the drive quickly, and it will be dead in no time. If this is your case, you could store only metadata, rather than the actual data block pages in the L2ARC. You can do this with the following:

```
# zfs set secondarycache=metadata pool
```

You can set this pool-wide, or per dataset. In the case outlined above, I would certainly do it pool-wide, which each dataset will inherit by default.

## Implementation

To this up, it's rather straight forward. Just identify what the drives are, by using their unique identifiers, then add them to the pool:

```
# ls /dev/disk/by-id/usb-* | grep -v part
/dev/disk/by-id/usb-Kingston_DataTraveler_G3_0014780D8CEBEC145E80163-0:0@
/dev/disk/by-id/usb-Kingston_DataTraveler_SE9_00187D0F567FEC2090007621-0:0@
/dev/disk/by-id/usb-Kingston_DataTraveler_SE9_00248121ABD5EC2070002E70-0:0@
```

```
/dev/disk/by-id/usb-Kingston_DataTraveler_SE9_00D0C9CE66A2EC2070002F04-0:0@
/dev/disk/by-id/usb-_USB_DISK_Pro_070B2605FA99D033-0:0@
/dev/disk/by-id/usb-_USB_DISK_Pro_070B2607A029C562-0:0@
/dev/disk/by-id/usb-_USB_DISK_Pro_070B2608976BFD58-0:0@
```

So, there are my seven drives that I outlined at the beginning of the post. So, to add them to the system as L2ARC drives, just run the following command:

```
# zpool add -f pool cache usb-Kingston_DataTraveler_G3_0014780D8CEBEBBC145E80163-0:0\
usb-Kingston_DataTraveler_SE9_00187D0F567FEC2090007621-0:0\
usb-Kingston_DataTraveler_SE9_00248121ABD5EC2070002E70-0:0\
usb-Kingston_DataTraveler_SE9_00D0C9CE66A2EC2070002F04-0:0\
usb-_USB_DISK_Pro_070B2605FA99D033-0:0\
usb-_USB_DISK_Pro_070B2607A029C562-0:0\
usb-_USB_DISK_Pro_070B2608976BFD58-0:0
```

Of course, these are the unique identifiers for my USB drives. Change them as necessary for your drives. Now that they are installed, are they filling up?

```
# zpool iostat -v
```

pool	alloc	free	read	write	read	write
pool	695G	1.13T	21	59	53.6K	457K
mirror	349G	579G	10	28	25.2K	220K
ata-ST1000DM003-9YN162_S1D1TM4J	-	-	4	21	25.8K	267K
ata-WDC_WD10EARS-00Y5B1_WD-WMAV50708780	-	-	4	21	27.9K	267K
mirror	347G	581G	11	30	28.3K	237K
ata-WDC_WD10EARS-00Y5B1_WD-WMAV50713154	-	-	4	22	16.7K	238K
ata-WDC_WD10EARS-00Y5B1_WD-WMAV50710024	-	-	4	22	19.4K	238K
logs	-	-	-	-	-	-
mirror	4K	1016M	0	0	0	0
ata-OCZ-REVODRIVE_OCZ-33W9WE11E9X73Y41-part1	-	-	0	0	0	0
ata-OCZ-REVODRIVE_OCZ-X5RG0E1Y7MN7676K-part1	-	-	0	0	0	0
cache	-	-	-	-	-	-
ata-OCZ-REVODRIVE_OCZ-33W9WE11E9X73Y41-part2	52.2G	16M	4	2	51.3K	291K
ata-OCZ-REVODRIVE_OCZ-X5RG0E1Y7MN7676K-part2	52.2G	16M	4	2	52.6K	293K

usb-Kingston_DataTraveler_G3_0014780D8CEBEB145E80163-0:0	465M	6.80G	0	0	319	72.8K
usb-Kingston_DataTraveler_SE9_00187D0F567FEC2090007621-0:0	1.02G	13.5G	0	0	1.58K	63.0K
usb-Kingston_DataTraveler_SE9_00248121ABD5EC2070002E70-0:0	1.17G	13.4G	0	0	844	72.3K
usb-Kingston_DataTraveler_SE9_00D0C9CE66A2EC2070002F04-0:0	990M	13.6G	0	0	1.02K	59.9K
usb-_USB_DISK_Pro_070B2605FA99D033-0:0	1.08G	6.36G	0	0	1.18K	67.0K
usb-_USB_DISK_Pro_070B2607A029C562-0:0	1.76G	5.68G	0	1	2.48K	109K
usb-_USB_DISK_Pro_070B2608976BFD58-0:0	1.20G	6.24G	0	0	530	38.8K

---

Something important to understand here, is the drives do not need to be all the same size. You can mix and match as you have on hand. Of course, the more space you can give to the cache, the better off you'll be.

## Conclusion

While this certainly isn't designed for speed, it can be used for lower random IO latencies, and it will reduce power in the datacenter. Further, what else are you going to do with those USB devices just lying around? Might as well put them to good use. Definitely seeing as though "the cloud" is making it trivial to get all of your files online.

Posted by Aaron Toponce on Thursday, May 9, 2013, at 6:00 am. Filed

under [Debian](#), [Linux](#), [Ubuntu](#), [ZFS](#). Follow any responses to this post

with its [comments RSS](#) feed. You can [post a comment](#) or [trackback](#) from

your blog. For IM, Email or Microblogs, here is the [Shortlink](#).

## { 4 } Comments

1. Jeremy Rosengren | May 9, 2013 at 8:03 am | [Permalink](#)

Question: It looks like you already have a couple of OCZ RevoDrives installed... in this particular scenario, do the USB cache devices still provide value? It does look like they're being used, does ZFS treat all the disks as the same