Navid Malek
navidmalekedu@gmail.com
navidmalek.blog.ir

Setting up LizardFS and testing it

objective:

- setup simple LizardFS master on one server
- test LizardFS performance on 3,5,7 chunk servers (HDD)
- test LizardFS performance on SSD

- LizardFS version ==> 3.12

Why LizardFS?

There are many benefits with LizardFS, some are :

- Easy to setup
- Easy to Scale
- Very robust against change of infrastructure
- Acceptable performance
- Open Source
- Active community and support

some debates:

- https://www.reddit.com/r/sysadmin/comments/5uulqm/best_distributed_file_system_glusterfs_vs_ceph_vs/
- https://www.jdieter.net/posts/2016/09/30/from-nfs-to-lizardfs/
- https://www.jdieter.net/posts/2017/08/14/benchmarking-small-file-performance-on-distributed-filesystems/

Navid Malek
navidmalekedu@gmail.com
navidmalek.blog.ir

Other good options:

- BeeGFS
- redhat’s Ceph Storage
- ZFS

Setup LizardFS

This is a very quick tutorial to setup LizardFS with default configuration:

https://www.dideo.ir/v/yt/LH_n8JToaGM

https://github.com/lizardfs/lizardfs/wiki/Quick-Start-Guide

the main document is here:

https://docs.lizardfs.com/adminguide/installation.html#

Testing LizardFS on HDD

The server configuration :

master ==> server 3

CGI server ==> server 6 //the dd command runs on server 6

two chunk servers ==> servers 4, 5

i have used «dd» command in order to measure I/O:

https://medium.com/@kenichishibata/test-i-o-performance-of-linux-using-dd-a5074f1de9ce

Navid Malek
navidmalekedu@gmail.com
navidmalek.blog.ir

first run, using two chunk servers

//mainly using an even number for chunk servers is not a good idea!

write performance

```
[root@Bench06 ~]# man dd
[root@Bench06 ~]# dd if=/dev/zero of=/mnt/lizardfs/test1.img bs=1G count=1 oflag=dsync
1+0 records in
1+0 records out
1073741824 bytes (1.1 GB) copied, 40.4214 s, 26.6 MB/s
[root@Bench06 ~]# dd if=/dev/zero of=/mnt/lizardfs/test2.img bs=10G count=1 oflag=dsync
0+1 records in
0+1 records out
2147479552 bytes (2.1 GB) copied, 76.9669 s, 27.9 MB/s
[root@Bench06 ~]#
```

Read Performance

```
[root@Bench06 lizardfs]# time dd if=/mnt/lizardfs/test2.img of=/dev/null bs=32k
65535+1 records in
65535+1 records out
2147479552 bytes (2.1 GB) copied, 7.74135 s, 277 MB/s

real    0m7.748s
user    0m0.041s
sys     0m1.320s
```

```
[root@Bench06 lizardfs]# time dd if=/mnt/lizardfs/test2.img of=/dev/null bs=64k
32767+1 records in
32767+1 records out
2147479552 bytes (2.1 GB) copied, 7.61881 s, 282 MB/s

real    0m7.625s
user    0m0.027s
```

```
[root@Bench06 lizardfs]# time dd if=/mnt/lizardfs/test2.img of=/dev/null
4194296+0 records in
4194296+0 records out
2147479552 bytes (2.1 GB) copied, 13.4009 s, 160 MB/s

real    0m13.407s
user    0m2.088s
sys     0m8.134s
```

Navid Malek
navidmalekedu@gmail.com
navidmalek.blog.ir

using three chunk servers

write

```
[root@Bench06 lizardfs]# dd if=/dev/zero of=/mnt/lizardfs/newtest1.img bs=1G count=1 oflag=dsync
1+0 records in
1+0 records out
1073741824 bytes (1.1 GB) copied, 41.4729 s, 25.9 MB/s
[root@Bench06 lizardfs]# dd if=/dev/zero of=/mnt/lizardfs/newtest2.img bs=2G count=1 oflag=dsync
0+1 records in
0+1 records out
2147479552 bytes (2.1 GB) copied, 80.9338 s, 26.5 MB/s
[root@Bench06 lizardfs]# dd if=/dev/zero of=/mnt/lizardfs/newtest2.img bs=10G count=1 oflag=dsync
0+1 records in
0+1 records out
2147479552 bytes (2.1 GB) copied, 82.2854 s, 26.1 MB/s
[root@Bench06 lizardfs]#
```

read

```
[root@Bench06 lizardfs]# echo 3 | sudo tee /proc/sys/vm/drop_caches
3
[root@Bench06 lizardfs]# time dd if=/mnt/lizardfs/newtest2.img of=/dev/null bs=32k
65535+1 records in
65535+1 records out
2147479552 bytes (2.1 GB) copied, 10.9108 s, 197 MB/s

real    0m10.917s
user    0m0.045s
sys     0m1.290s
[root@Bench06 lizardfs]# time dd if=/mnt/lizardfs/newtest2.img of=/dev/null bs=128k
16383+1 records in
16383+1 records out
2147479552 bytes (2.1 GB) copied, 0.613303 s, 3.5 GB/s

real    0m0.617s
user    0m0.008s
sys     0m0.606s
[root@Bench06 lizardfs]# echo 3 | sudo tee /proc/sys/vm/drop_caches
3
[root@Bench06 lizardfs]# time dd if=/mnt/lizardfs/newtest2.img of=/dev/null bs=128k
16383+1 records in
16383+1 records out
2147479552 bytes (2.1 GB) copied, 7.84701 s, 274 MB/s

real    0m7.853s
user    0m0.025s
sys     0m1.183s
```

Navid Malek
navidmalekedu@gmail.com
navidmalek.blog.ir

```
[root@Bench06 lizardfs]# ^C
[root@Bench06 lizardfs]# dd if=/dev/zero of=/tmp/ttime dd if=/mnt/lizardfs/test2.img of=/dev/null bs=64^Cstlatency2.img bs
=512 count=5000 oflag=dsync
[root@Bench06 lizardfs]# time dd if=/mnt/lizardfs/newtest2.img of=/dev/null bs=64k
32767+1 records in
32767+1 records out
2147479552 bytes (2.1 GB) copied, 0.485367 s, 4.4 GB/s

real    0m0.490s
user    0m0.005s
sys     0m0.477s
[root@Bench06 lizardfs]# echo 3 | sudo tee /proc/sys/vm/drop_caches
3
[root@Bench06 lizardfs]# time dd if=/mnt/lizardfs/newtest2.img of=/dev/null bs=64k
32767+1 records in
32767+1 records out
2147479552 bytes (2.1 GB) copied, 11.5099 s, 187 MB/s

real    0m11.516s
user    0m0.022s
sys     0m1.164s
[root@Bench06 lizardfs]# echo 3 | sudo tee /proc/sys/vm/drop_caches
3
[root@Bench06 lizardfs]# time dd if=/mnt/lizardfs/newtest2.img of=/dev/null bs=32k
65535+1 records in
65535+1 records out
2147479552 bytes (2.1 GB) copied, 10.9108 s, 197 MB/s
```

using 5 chunk servers

write

```
[root@Bench06 lizardfs-meta]# dd if=/dev/zero of=/mnt/lizardfs/newtest1.img bs=1G count=1 oflag=
dsync
1+0 records in
1+0 records out
1073741824 bytes (1.1 GB) copied, 44.3508 s, 24.2 MB/s
[root@Bench06 lizardfs-meta]# dd if=/dev/zero of=/mnt/lizardfs/newtest1.img bs=2G count=1 oflag=
dsync
0+1 records in
0+1 records out
2147479552 bytes (2.1 GB) copied, 84.7874 s, 25.3 MB/s
[root@Bench06 lizardfs-meta]# dd if=/dev/zero of=/mnt/lizardfs/newtest2.img bs=2G count=1 oflag=
dsync
0+1 records in
0+1 records out
2147479552 bytes (2.1 GB) copied, 85.8915 s, 25.0 MB/s
[root@Bench06 lizardfs-meta]# dd if=/dev/zero of=/mnt/lizardfs/newtest1.img bs=1G count=1 oflag=
dsync
1+0 records in
1+0 records out
1073741824 bytes (1.1 GB) copied, 41.52 s, 25.9 MB/s
```

Navid Malek
navidmalekedu@gmail.com
navidmalek.blog.ir

Read

```
[root@Bench06 lizardfs-meta]# echo 3 | sudo tee /proc/sys/vm/drop_caches ; time dd if=/mnt/lizar
dfs/newtest2.img of=/dev/null bs=16k
3
131071+1 records in
131071+1 records out
2147479552 bytes (2.1 GB) copied, 13.2539 s, 162 MB/s

real    0m13.260s
user    0m0.091s
sys     0m1.660s
[root@Bench06 lizardfs-meta]# echo 3 | sudo tee /proc/sys/vm/drop_caches ; time dd if=/mnt/lizardfs/newtest2.img
 of=/dev/null bs=32k
3
65535+1 records in
65535+1 records out
2147479552 bytes (2.1 GB) copied, 7.56776 s, 284 MB/s

real    0m7.574s
user    0m0.038s
sys     0m1.292s
[root@Bench06 lizardfs-meta]# echo 3 | sudo tee /proc/sys/vm/drop_caches ; time dd if=/mnt/lizardfs/newtest2.img
 of=/dev/null bs=64k
3
32767+1 records in
32767+1 records out
2147479552 bytes (2.1 GB) copied, 7.69656 s, 279 MB/s
```

using 7 chunk servers

write

```
[root@Bench06 trash]# dd if=/dev/zero of=/mnt/lizardfs/newtest1.img bs=1G count=1 oflag=dsync
1+0 records in
1+0 records out
1073741824 bytes (1.1 GB) copied, 44.0598 s, 24.4 MB/s
[root@Bench06 trash]# dd if=/dev/zero of=/mnt/lizardfs/newtest2.img bs=2G count=1 oflag=dsync
0+1 records in
0+1 records out
2147479552 bytes (2.1 GB) copied, 89.3548 s, 24.0 MB/s
[root@Bench06 trash]# dd if=/dev/zero of=/mnt/lizardfs/newtest3.img bs=2G count=1 oflag=dsync
^[[15~0+1 records in
0+1 records out
2147479552 bytes (2.1 GB) copied, 85.828 s, 25.0 MB/s
[root@Bench06 trash]# dd if=/dev/zero of=/mnt/lizardfs/newtest4.img bs=2G count=1 oflag=dsync
0+1 records in
0+1 records out
2147479552 bytes (2.1 GB) copied, 86.3868 s, 24.9 MB/s
[root@Bench06 trash]#
```

Read

Navid Malek
navidmalekedu@gmail.com
navidmalek.blog.ir

```
[root@Bench06 trash]# echo 3 | sudo tee /proc/sys/vm/drop_caches ; time dd if=/mnt/lizardfs/newtest4.img of=/dev
/null bs=32k
3
65535+1 records in
65535+1 records out
2147479552 bytes (2.1 GB) copied, 7.96675 s, 270 MB/s

real    0m7.973s
user    0m0.038s
sys     0m1.320s
[root@Bench06 trash]# echo 3 | sudo tee /proc/sys/vm/drop_caches ; time dd if=/mnt/lizardfs/newtest4.img of=/dev
/null bs=64k
3
32767+1 records in
32767+1 records out
2147479552 bytes (2.1 GB) copied, 7.56979 s, 284 MB/s

real    0m7.577s
user    0m0.023s
sys     0m1.273s
[root@Bench06 trash]# echo 3 | sudo tee /proc/sys/vm/drop_caches ; time dd if=/mnt/lizardfs/newtest4.img of=/dev
/null bs=128k
3
16383+1 records in
16383+1 records out
2147479552 bytes (2.1 GB) copied, 7.13805 s, 301 MB/s
```

Summery

As you see performance is not what we expected from HDD and SSD, for example in the first scenario :

- 1 master/CGI server
- 2 chunk servers

HDD servers:

read ==> 270 MB/s
write ==> 25 MB/s

SSD servers:

read ==> around 36 MB/s
write ==> around 16 MB/s

Navid Malek
navidmalekedu@gmail.com
navidmalek.blog.ir

troubleshooting

Suggestions from Jonathan Dieter (https://www.jdieter.net)

enabling/disabling PERFORM_FSYNC

```
navidx@navidx-ThinkPad-E460:~$ ssh root@172.30.18.26
Last login: Mon Oct  8 09:07:44 2018 from 172.30.30.179
[root@Bench06 ~]# dd if=/dev/zero of=/home/verifytest2.img bs=2G count=1 oflag=dsync
0+1 records in
0+1 records out
2147479552 bytes (2.1 GB) copied, 13.6706 s, 157 MB/s
[root@Bench06 ~]# dd if=/dev/zero of=/mnt/lizardfs/verifytest2.img bs=2G count=1 oflag=dsync
0+1 records in
0+1 records out
2147479552 bytes (2.1 GB) copied, 82.9119 s, 25.9 MB/s
[root@Bench06 ~]# dd if=/dev/zero of=/mnt/lizardfs/verifytest3.img bs=2G count=1 oflag=dsync
0+1 records in
0+1 records out
2147479552 bytes (2.1 GB) copied, 89.8427 s, 23.9 MB/s
[root@Bench06 ~]#
```

first command ==> directly on local disk
second command ==> lizardfs FSYNC enabled
third command ==> lizardfs FSYNC disabled

there is not much of difference, so the problem should be elsewhere.

Suggestions from Szymon Haly (CEO of LizardFS)

enabling BIG_WRITE option on mfsmount:

SSD servers:
write ==> 220 MB/s
read ==> 20 MB/s

HDD servers:
write ==> 134 MB/s
read ==> 250 MB/s

The writes are much better, but as you see the read speed of SSD servers is still very low.

Navid Malek
navidmalekedu@gmail.com
navidmalek.blog.ir

Another option was to consider the EC replication instead of default replications.

To better understand EC replica see this page:

https://docs.lizardfs.com/adminguide/replication.html

in simple words, EC replica breaks the data into n parts and m parity parts and distribute them on n + m chunk servers, this technique helps to better achieve parallel read/writes because of distribution of data parts.

First add the new replica in the mfsgoals.cfg:

```
1 1 : _
2 2 : _ _
3 3 : _ _ _
4 4 : _ _ _ _
5 5 : _ _ _ _ _
6 ec_config : $ec(2,1)
```

Next add the goal to the lizardfs mount directory

https://docs.lizardfs.com/adminguide/replication.html#show-current-goal-configuration

now if you have 3 chunk servers, each data part of a big data will be divided into two parts and one parity part and resides on three separate serves.

//TO DO

the test results are not available at the moment, in order to test SSD cluster we have to support a 1000 capacity network switch in our local network. if we don't, the network switch rate will be the bottleneck (maximum rate is around 10 Mbps) and we cannot actually measure the performance of LizardFS in local network.