

Hard Drive Burn in Testing

<https://www.truenas.com/community/resources/hard-drive-burn-in-testing.92/>

TESTING FOR HIGH CAPACITY DRIVES WILL TAKE A LONG TIME. EXPECT 12 HOURS OR SO PER TB OF CAPACITY

First of all, the S.M.A.R.T. tests. The first thing that someone unfamiliar with S.M.A.R.T. tests might find strange is the fact that no results are shown when you run the test. The way these tests work is that you initiate the test, it goes off and does its thing, then it records the results for you to check later. So, if this is an initial burn-in test for your entire system, you can initiate tests on all of the drives simultaneously by simply issuing the test command for each drive one after another.

The first test to run is a short self-test:

Code:

```
smartctl -t short /dev/adaX
```

It should indicate that the test will take about 5 minutes. You can immediately begin the same test on the next drive, but you can only run one test on each drive at a time. Once it has completed, run a conveyance test:

Code:

```
smartctl -t conveyance /dev/adaX
```

Again, wait for the test to complete (about 2 minutes this time). Finally, a long test:

Code:

```
smartctl -t long /dev/adaX
```

Note added by [@wblock](#) 2018-01-10: this section recommended enabling the `kern.geom.debugflags sysctl`. Many people still think it has something to do with allowing raw writes. It does not. Instead, it disables a safety system that is intended to prevent writes to disks that are in use (say, by having a mounted filesystem). From [man 4 geom](#):

0x10 (allow foot shooting)

Allow writing to Rank 1 providers. This would, for example, allow the super-user to overwrite the MBR on the root disk or write random sectors elsewhere to a mounted disk. The implications are obvious.

To summarize, this option should generally not be needed. It only makes it possible to harm data. Any disk you are going to overwrite with data should not be mounted or have anything you wish to keep. In fact, best practice is to not be erasing or stress-testing drives on a system that has actual data on it. Since those disks will not have mounted filesystems, this sysctl will not affect being able to write to them. In fact, it will only make it possible to blow away things that are in use.

Now, before we can perform raw disk I/O, we need to enable the kernel geometry debug flags.

This carries some inherent risk, and should probably not be done on a production system. This does not survive through a reboot, so when you're done, just reboot the machine to disable it:

Code:

```
sysctl kern.geom.debugflags=0x10
```

Now that we can execute raw I/O, run a badblocks r/w test.

Unlike the S.M.A.R.T. tests, badblocks runs in the foreground, so once you start it, you won't be able to use the console until the test completes. It also means that if you start it over SSH and lose your connection, the test will be canceled. The answer to this is to use a utility called tmux:

Code:

```
tmux
```

You should now see a green stripe at the bottom of the screen. Now, we can run badblocks. **THIS TEST WILL DESTROY ANY DATA ON THE DISK SO ONLY RUN THIS ON A NEW DISK WITHOUT DATA ON IT OR BACK UP ANY DATA FIRST:**

Code:

```
badblocks -ws /dev/adaX
```

badblocks also offers a non-destructive read-write test that (in theory) shouldn't damage any existing data, but if you do choose to run it on a production drive and suffer data loss, on your own head be it:

Code:

```
badblocks -ns /dev/adaX
```

It has been brought to my attention that badblocks has some limitations with larger drives >2TB. The easy workaround is to manually specify a larger block size for the test.

Code:

```
badblocks -b 4096 -ws /dev/adaX
```

or

Code:

```
badblocks -b 4096 -ns /dev/adaX
```

Once you've started the first test, press Ctrl+B, then " (the double-quote key, not the single quote twice). You should now see a half-white, half-green line through the screen (in PuTTY, it's q's instead of a line, but same thing) with the test continuing in the top half of the screen and a new shell prompt in the bottom. Run the badblocks command again on the next disk, then press Ctrl+B, " again to create another shell. Continue until you've started a test on each disk. If you are connecting over SSH and your session gets disconnected, all of the tests will continue running. When you reconnect, to resume the session and view the test status, simply type:

Code:

```
tmux attach
```

As with the S.M.A.R.T. tests, you can only run one test at a time per drive, but you can test all of your drives simultaneously. In my experience, the tests run just as fast with all drives testing as with a single drive, so for your initial burn-in, there's really no reason not to test all of the drives at once. Also, be prepared for this test to take a very long time, as it is basically the "meat and potatoes" of your burn-in process. For reference, the default 4-pass r/w test took a little over 24 hours on my WD Red 2TB drives, YMMV.

Because S.M.A.R.T. tests only passively detect errors after you've actually attempted to read or write a bad sector, you should run the S.M.A.R.T. long test again after badblocks completes:

Code:

```
smartctl -t long /dev/adaX
```

At this point, you have fully tested all of your drives, and now it's time to view the results of the various S.M.A.R.T. tests:

Code:

```
smartctl -A /dev/adaX
```

This should produce something like this (sorry for the formatting fail):

Code:

```
[root@freenas] ~# smartctl -A /dev/ada0
smartctl 6.2 2013-07-26 r3841 [FreeBSD 9.2-RELEASE-p4 amd64] (local build)
Copyright (C) 2002-13, Bruce Allen, Christian Franke, www.smartmontools.org

=== START OF READ SMART DATA SECTION ===
SMART Attributes Data Structure revision number: 16
Vendor Specific SMART Attributes with Thresholds:
ID# ATTRIBUTE_NAME          FLAG     VALUE WORST THRESH TYPE      UPDATED  WHEN_FAILED RAW_VALUE
 1 Raw_Read_Error_Rate     0x002f   200   200   051     Pre-fail Always    -        0
 3 Spin_Up_Time            0x0027   175   174   021     Pre-fail Always    -       4208
 4 Start_Stop_Count        0x0032   100   100   000     Old_age  Always    -        9
 5 Reallocated_Sector_Ct   0x0033   200   200   140     Pre-fail Always    -        0
 7 Seek_Error_Rate         0x002e   200   200   000     Old_age  Always    -        0
 9 Power_On_Hours          0x0032   100   100   000     Old_age  Always    -       357
10 Spin_Retry_Count        0x0032   100   253   000     Old_age  Always    -        0
11 Calibration_Retry_Count 0x0032   100   253   000     Old_age  Always    -        0
12 Power_Cycle_Count       0x0032   100   100   000     Old_age  Always    -        9
192 Power-Off_Retract_Count 0x0032   200   200   000     Old_age  Always    -        4
193 Load_Cycle_Count       0x0032   200   200   000     Old_age  Always    -        9
194 Temperature_Celsius    0x0022  119   113   000     Old_age  Always    -       28
196 Reallocated_Event_Count 0x0032   200   200   000     Old_age  Always    -        0
197 Current_Pending_Sector 0x0032   200   200   000     Old_age  Always    -        0
198 Offline_Uncorrectable   0x0030   100   253   000     Old_age  Offline   -        0
199 UDMA_CRC_Error_Count    0x0032   200   200   000     Old_age  Always    -        0
200 Multi_Zone_Error_Rate   0x0008   200   200   000     Old_age  Offline   -        0
```

Some of the more important fields right now include the `Reallocated_Sector_Ct`, `Current_Pending_Sector`, and `Offline_Uncorrectable` lines. All of these should have a `RAW_VALUE` of 0. I'm not sure why the `VALUE` field is listed as 200, but as long as the `RAW_VALUE` for each of these fields is 0, that means there are currently no bad sectors. Any result greater than 0 on a new drive should be cause for an immediate RMA.

Once all of your tests have completed, you should reboot your system to disable the kernel geometry debug flags.

Revision #5

Created 22 December 2023 02:15:28 by ColtM

Updated 7 August 2024 23:24:39 by ColtM