Monitoring Hard Disks with SMART

One of your hard disks might be trying to tell you it's not long for this world. Install software that lets you know when to replace it. **BY BRUCE ALLEN**

It's a given that all disks eventually die, and it's easy to see why. The platters in a modern disk drive rotate more than a hundred times per second, maintaining submicron tolerances between the disk heads and the magnetic media that store data. Often they run 24/7 in dusty, overheated environments, thrashing on heavily loaded or poorly managed machines. So, it's not surprising that experienced users are all too familiar with the symptoms of a dying disk. Strange things start happening. Inscrutable kernel error messages cover the console and then the system becomes unstable and locks up. Often, entire days are lost repeating recent work, re-installing the OS and trying to recover data. Even if you have a recent backup, sudden disk failure is a minor catastrophe.

Many users and system administrators don't know that Self-Monitoring, Analysis and Reporting Technology systems (SMART) are built in to most modern ATA and SCSI hard disks. SMART disk drives internally monitor their own health and performance. In many cases, the disk itself provides advance warning that something is wrong, helping to avoid the scenario described above. Most implementations of SMART also allow users to perform self-tests on the disk and to monitor a number of performance and reliability attributes.

By profession I am a physicist. My research group runs a large computing cluster with 300 nodes and 600 disk drives, on which more than 50TB of physics data are stored. I became interested in SMART several years ago when I realized it could help reduce downtime and keep our cluster operating more reliably. For about a year I have been maintaining an open-source package called smartmontools, a spin-off of the UCSC smartsuite package, for this purpose.

In this article, I explain how to use smartmontools' smartctl utility and smartd dæmon to monitor the health of a system's disks. See smartmontools.sourceforge.net for download and installation instructions and consult the WARNINGS file for a list of problem disks/controllers. Additional documentation can be found in the man pages (`man smartctl` and `man smartd`) and on the Web page.

Versions of smartmontools are available for Slackware, Debian, SuSE, Mandrake, Gentoo, Conectiva and other Linux distributions. Red Hat's existing products contain the UCSC smartsuite versions of smartctl and smartd, but the smartmontools versions will be included in upcoming releases.

To understand how smartmontools works, it's helpful to know the history of SMART. The original SMART spec (SFF-8035i) was written by a group of disk drive manufacturers. In Revision 2 (April 1996) disks keep an internal list of up to 30 Attributes corresponding to different measures of performance and reliability, such as read and seek error rates. Each Attribute has a one-byte normalized value ranging from 1 to 253 and a corresponding one-byte threshold. If one or more of the normalized Attribute values are less than or equal to its corresponding threshold, then either the disk is expected to fail in less than

Listing 1. Output of `smartctl -i /dev/hda`

```
Device Model:     IC35L120AVV207-0
Serial Number:    VNVD02G4G3R72G
Firmware Version: V240A63A
Device is:        In smartctl database [for details use: -P show]
ATA Version is:   6
ATA Standard is:  ATA/ATAPI-6 T13 1410D revision 3a
SMART support is: Available - device has SMART capability.
SMART support is: Enabled
```

Listing 2. Output of `smartctl -Hc /dev/hda`

```
SMART overall-health self-assessment test result: PASSED

General SMART Values:
Off-line data collection status: (0x82)  Offline data collection activity
                                         was completed without error.
                                         Auto Off-line Data Collection:
Enabled.
Self-test execution status:      (   0)  The previous self-test routine
                                         completed without error or no
                                         self-test has ever been run.
Total time to complete off-line
data collection:                 (2855)  seconds.
Offline data collection
capabilities:                    (0x1b)  SMART execute Offline immediate.
                                         Automatic timer ON/OFF support.
                                         Suspend Offline collection upon new
                                         command.
                                         Offline surface scan supported.
                                         Self-test supported.
                                         No Conveyance Self-test supported.
                                         No Selective Self-test supported.
SMART capabilities:            (0x0003)  Saves SMART data before entering
                                         power-saving mode.
                                         Supports SMART auto save timer.
Error logging capability:        (0x01)  Error logging supported.
                                         General Purpose Logging supported.
Short self-test routine
recommended polling time:        (   1)  minutes.
Extended self-test routine
recommended polling time:        (  48)  minutes.
```

Listing 3. Output of `smartctl -A /dev/hda`

```
Vendor Specific SMART Attributes with Thresholds:
ID# ATTRIBUTE_NAME          FLAG     VALUE WORST THRESH TYPE      UPDATED  WHEN_FAILED RAW_VALUE
  1 Raw_Read_Error_Rate     0x000b   100   100   060    Pre-fail  Always       -       0
  2 Throughput_Performance  0x0005   155   155   050    Pre-fail  Offline      -       225
  3 Spin_Up_Time            0x0007   097   097   024    Pre-fail  Always       -       293 (Average 270)
  4 Start_Stop_Count        0x0012   100   100   000    Old_age   Always       -       10
  5 Reallocated_Sector_Ct   0x0033   100   100   005    Pre-fail  Always       -       0
  7 Seek_Error_Rate         0x000b   100   100   067    Pre-fail  Always       -       0
  8 Seek_Time_Performance   0x0005   125   125   020    Pre-fail  Offline      -       36
  9 Power_On_Hours          0x0012   100   100   000    Old_age   Always       -       3548
 10 Spin_Retry_Count        0x0013   100   100   060    Pre-fail  Always       -       0
 12 Power_Cycle_Count       0x0032   100   100   000    Old_age   Always       -       10
192 Power-Off_Retract_Count 0x0032   100   100   050    Old_age   Always       -       158
193 Load_Cycle_Count        0x0012   100   100   050    Old_age   Always       -       158
194 Temperature_Celsius     0x0002   189   189   000    Old_age   Always       -       29 (Lifetime Min/Max 23/33)
196 Reallocated_Event_Count 0x0032   100   100   000    Old_age   Always       -       0
197 Current_Pending_Sector  0x0022   100   100   000    Old_age   Always       -       0
198 Offline_Uncorrectable   0x0008   100   100   000    Old_age   Offline      -       0
199 UDMA_CRC_Error_Count    0x000a   200   200   000    Old_age   Always       -       0
```

24 hours or it has exceeded its design or usage lifetime. Some of the Attribute values are updated as the disk operates. Others are updated only through off-line tests that temporarily slow down disk reads/writes and, thus, must be run with a special command. In late 1995, parts of SFF-8035i were merged into the ATA-3 standard.

Starting with the ATA-4 standard, the requirement that disks maintain an internal Attribute table was dropped. Instead, the disks simply return an OK or NOT OK response to an inquiry about their health. A negative response indicates the disk firmware has determined that the disk is likely to fail. The ATA-5 standard added an ATA error log and commands to run disk self-tests to the SMART command set.

To make use of these disk features, you need to know how to use smartmontools to examine the disk's Attributes (most disks are backward-compatible with SFF-8035i), query the disk's health status, run disk self-tests, examine the disk's self-test log (results of the last 21 self-tests) and examine the disk's ATA error log (details of the last five disk errors). Although this article focuses on ATA disks, additional documentation about SCSI devices can be found on the smartmontools Web page.

To begin, give the command `smartctl -a /dev/hda`, using the correct path to your disk, as root. If SMART is not enabled on the disk, you first must enable it with the `-s on` option. You then see output similar to the output shown in Listings 1–5.

The first part of the output (Listing 1) lists model/firmware information about the disk—this one is an IBM/Hitachi GXP-180 example. smartmontools has a database of disk types. If your disk is in the database, it may be able to interpret the raw Attribute values correctly.

The second part of the output (Listing 2) shows the results of the health status inquiry. This is the one-line Executive Summary Report of disk health; the disk shown here has passed. If your disk health status is FAILING, back up your data immediately. The remainder of this section of the output provides information about the disk's capabilities and the estimated time to perform short and long disk self-tests.

The third part of the output (Listing 3) lists the disk's table of up to 30 Attributes (from a maximum set of 255). Remember that Attributes are no longer part of the ATA standard, but most manufacturers still support them. Although SFF-8035i doesn't define the meaning or interpretation of Attributes, many have a de facto standard interpretation. For example, this disk's 13th Attribute (ID #194) tracks its internal temperature.

Studies have shown that lowering disk temperatures by as little as 5°C significantly reduces failure rates, though this is less of an issue for the latest generation of fluid-drive bearing drives. One of the simplest and least expensive steps you can take to ensure disk reliability is to add a cooling fan that blows cooling air directly onto or past the system's disks.

Each Attribute has a six-byte raw value (RAW_VALUE) and a one-byte normalized value (VALUE). In this case, the raw value stores three temperatures: the disk's temperature in Celsius (29), plus its lifetime minimum (23) and maximum (33) values. The format of the raw data is vendor-specific and not specified by any standard. To track disk reliability, the disk's firmware converts the raw value to a normalized value ranging from 1 to 253. If this normalized value is less than or equal to the threshold (THRESH), the Attribute is said to have failed, as indicated in the WHEN_FAILED column. The column is empty because none of these Attributes has failed. The lowest (WORST) normalized value also is shown; it is the smallest value attained since SMART was enabled on the disk. The TYPE of the Attribute indicates if Attribute failure means the device has reached the end of its design life (Old_age) or it's an impending disk failure (Pre-fail). For example, disk spin-up time (ID #3) is a prefailure Attribute. If this (or any other prefail Attribute) fails, disk failure is predicted in less than 24 hours.

The names/meanings of Attributes and the interpretation of their raw values is not specified by any standard. Different manufacturers sometimes use the same Attribute ID for different purposes. For this reason, the interpretation of specific Attributes can be modified using the `-v` option to `smartctl`; please see the man page for details. For example, some disks use Attribute 9 to store the power-on time of the disk in minutes; the `-v 9,minutes` option to `smartctl` correctly modifies the Attribute's interpretation. If your disk model is in the smartmontools database, these -v options are set automatically.

The next part of the `smartctl -a` output (Listing 4) is a log of the disk errors. This particular disk has been error-free, and

The SMART standard provides a mechanism for running disk self-tests and for monitoring aspects of disk performance.

the log is empty. Typically, one should worry only if disk errors start to appear in large numbers. An occasional transient error that does not recur usually is benign. The smartmontools Web page has a number of examples of `smartctl -a` output showing some illustrative error log entries. They are timestamped with the disk's power-on lifetime in hours when the error occurred, and the individual ATA commands leading up to the error are timestamped with the time in milliseconds after the disk was powered on. This shows whether the errors are recent or old.

The final part of the `smartctl` output (Listing 5) is a report of the self-tests run on the disk. These show two types of self-tests, short and long. (ATA-6/7 disks also may have conveyance and selective self-tests.) These can be run with the commands `smartctl -t short /dev/hda` and `smartctl -t long /dev/hda` and do not corrupt data on the disk. Typically, short tests take only a minute or two to complete, and long tests take about an hour. These self-tests do not interfere with the normal functioning of the disk, so the commands may be used for mounted disks on a running system. On our computing cluster nodes, a long self-test is run with a cron job early every Sunday morning. The entries in Listing 5 all are self-tests that completed without errors; the LifeTime column shows the power-on age of the disk when the self-test was run. If a self-test finds an error, the Logical Block Address (LBA) shows where the error occurred on the disk. The Remaining column shows the percentage of the self-test remaining when the error was found. If you suspect that something is wrong with a disk, I strongly recommend running a long self-test to look for problems.

The `smartctl -t offline` command can be used to carry out off-line tests. These off-line tests do not make entries in the self-test log. They date back to the SFF-8035i standard, and update values of the Attributes that are not updated automatically under normal disk operation (see the UPDATED column in Listing 3). Some disks support automatic off-line testing, enabled by `smartctl -o on`, which automatically runs an off-line test every few hours.

Listing 4. Output of `smartctl -l error /dev/hda`

```
SMART Error Log Version: 1
No Errors Logged
```

Listing 5. Output of `smartctl -l selftest /dev/hda`

```
SMART Self-test log, version number 1
Num  Test_Description    Status       Remaining  LifeTime(hours)  LBA_of_first_error
# 1  Extended off-line   Completed    00%        3525             -
# 2  Extended off-line   Completed    00%        3357             -
# 3  Short off-line      Completed    00%        3059             -
```

The SMART standard provides a mechanism for running disk self-tests and for monitoring aspects of disk performance. Its main shortcoming is that it doesn't provide a direct mechanism for informing the OS or user if problems are found. In fact, because disk SMART status frequently is not monitored, many disk problems go undetected until they lead to catastrophic failure. Of course, you can monitor disks on a regular basis using the smartctl utility, as I've described, but this is a nuisance.

The remaining part of the smartmontools package is the smartd dæmon that does regular monitoring for you. It monitors the disk's SMART data for signs of problems. It can be configured to send e-mail to users or system administrators or to run arbitrary scripts if problems are detected. By default, when smartd is started, it registers the system's disks. It then checks their status every 30 minutes for failing Attributes, failing health status or increased numbers of ATA errors or failed self-tests and logs this information with SYSLOG in /var/log/messages by default.

You can control and fine-tune the behavior of smartd using the configuration file /etc/smartd.conf. This file is read when smartd starts up, before it forks into the background. Each line contains Directives pertaining to a different disk. The configuration file on our computing cluster nodes look like this:

```
# /etc/smartd.conf config file
/dev/hda -S on -o on -a -I 194 -m sense@phys.uwm.edu
/dev/hdc -S on -o on -a -I 194 -m sense@phys.uwm.edu
```

The first column indicates the device to be monitored. The `-o on` Directive enables the automatic off-line testing, and the `-S on` Directive enables automatic Attribute autosave. The `-m` Directive is followed by an e-mail address to which warning messages are sent, and the `-a` Directive instructs smartd to monitor all SMART features of the disk. In this configuration, smartd logs changes in all normalized attribute values. The `-I 194` Directive means ignore changes in Attribute #194, because disk temperatures change often, and it's annoying to have such changes logged on a regular basis.

Normally smartd is started by the normal UNIX init mechanism. For example, on Red Hat distributions, `/etc/rc.d/init.d/smartd start` and `/etc/rc.d/init.d/smartd stop` can be used to start and stop the dæmon.

Further information about the smartd and its config file can be found in the man page (`man smartd`), and summaries can be found with the commands `smartd -D` and `smartd -h`. For example, the `-M test` Directive sends a test e-mail warning message to confirm that warning e-mail messages are delivered correctly. Other Directives provide additional flexibility, such as monitoring changes in raw Attribute values.

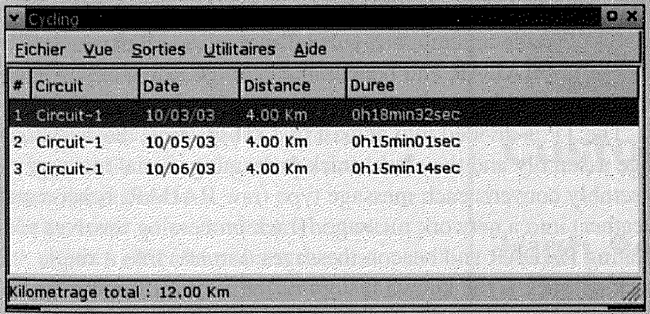
What should you do if a disk shows signs of problems? What if a disk self-test fails or the disk's SMART health status fails? Start by getting your data off the disk and on to another system as soon as possible. Second, run some extended disk self-tests and see if the problem is repeatable at the same LBA. If so, something probably is wrong with the disk. If the disk has failing SMART health status and is under warranty, the vendor usually will replace it. If the disk is failing its self-tests, many manufacturers provide specialized disk health programs, for example, Maxtor's PowerMax or IBM's Drive Fitness Test. Sometimes these programs actually can repair a disk by remapping bad sectors. Often, they report a special error code that can be used to get a replacement disk.

This article has covered the basics of smartmontools. To learn more, read the man pages and Web page, and then write to the support mailing list if you need further help. Remember, smartmontools is no substitute for backing up your data. SMART cannot and does not predict all disk failures, but it often provides clues that something is amiss and has helped to keep many computing clusters operating reliably.

Several developers are porting smartmontools to FreeBSD, Darwin and Solaris, and we recently have added extensions to allow smartmontools to monitor and control the ATA disks behind 3ware RAID controllers. If you would like to contribute to the development of smartmontools, write to the support mailing list. We especially are interested in information about the interpretation and meaning of vendor-specific SMART Attribute and raw values.■

Bruce Allen is a professor of Physics at the University of Wisconsin - Milwaukee. He does research work on gravitational waves and the very early universe, and he has built several large Linux clusters for data analysis use.

Cycling: juvenis.dyndns.org/index.php?dirid=1&artid=3



#	Circuit	Date	Distance	Duree
1	Circuit-1	10/03/03	4.00 Km	0h18min32sec
2	Circuit-1	10/05/03	4.00 Km	0h15min01sec
3	Circuit-1	10/06/03	4.00 Km	0h15min14sec

Kilometrage total : 12.00 Km

Cycling is designed to help you with your cycling training, by keeping track of courses run, maximum speeds, distances and more. In fact, although designed for cycling, there's no reason it can't be put to use for running, swimming or almost any sport in which you might want to record time, distance and speed. Requires: libgtk-x11-2.0, libatk-1.0, libgdk-x11-2.0, libgdk_pixbuf-2.0, libm, libpangoxft-1.0, libpangox-1.0, libpango-1.0, libgobject-2.0, libgmodule-2.0, libdl, libglib-2.0, glibc, libXi, libXext, libXft, libXrender, libfontconfig, libX11, libfreetype, libz and libexpat.

—DAVID A. BANDEL