

Software Erasure of Hard Drives and ExpertEraser Technical Description

1. Introduction.

Today we live in the information age. This means that information is a commodity, and as such must be protected. There are two fundamental ways that information can become a serious threat, namely loss of information, and uncontrolled spreading of information. This paper will discuss software erasure as a method to eliminate the threat associated with unintentional spreading of information stored on hard drives.

So what does it mean that a drive is securely erased?

In our opinion, it means that it is impossible for an attacker to recover any data from any area of the hard drive. An attacker should not be able to deduce anything about the previous data stored on the drive, except that it is now erased.

The next question then is how do we achieve this?

As stated, we believe that erasing a drive with the proper high quality software is a good solution. To justify this we must first look at what goes on inside a hard drive.

If you are satisfied that overwritten data cannot be recovered, you may skip to section 3, "Software Requirements".

2. Hard Drive Technology

In the context of data erasure there are two aspects of a hard drive that are important:

- Physics and methods involved in the write and read processes
- How the drive physically accesses the data

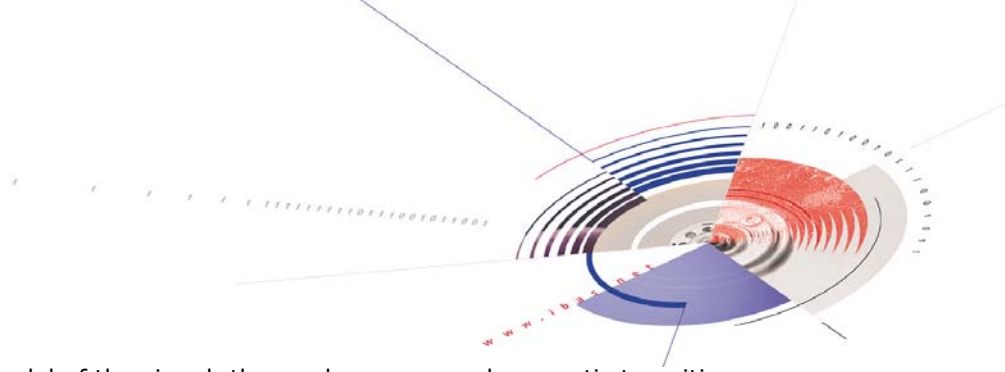
The physics of the erase process

This section will try to describe the physical process of erasing data by discussing the 'normal' read and write operations of a drive.

Magnetic recording relies on a material property called magnetic remanence. This could be thought of as a material's ability to 'remember' the direction of its last magnetization. All hard drives on the market today (2000) utilize something called saturation recording. This can be thought of as a digital process in the sense that the magnetic media (the platters) are fully magnetized in one of two directions (N-S or S-N). Thus the process of writing is a matter of magnetizing small areas of the platters in one of these two directions. It is also important to note that the write process is non-linear (as dictated by the hysteresis curve).

Modern hard drives use direct overwrite, which means that new data is written directly over the old data. Thus there is no erasing prior to writing as there is for some magnetic tape systems.

Two very important aspects of the reading process is the read process model and the signal to noise ratio in the process.



Model: A drive uses a theoretical model of the signal, the read process, and magnetic transition interactions to read data. This model represents what the disk drive expects the data signal to look like (this method is called PRML=Partial Response Maximum Likelihood). All modern hard drives use this or a similar linear model in the read process. We said earlier that the write process is non-linear, so, you might ask, how does this match the linear read model assumption? It turns out that the combination of something called pre-compensation and saturation recording make the linear model fairly accurate (for moderate signal to noise ratios).

Signal to Noise Ratio (SNR): The SNR tells us the relative strengths of the signal (the data), and unwanted noise. Information Theory tells us that there is an upper limit to how much data we can store and reliably retrieve at a given SNR. However, advances in storage technology have made it possible for drives to operate very close to this limit.

With this in mind, let's look at what happens when we overwrite old data with new data. Because of the saturation recording technique, the effect of overwriting old data will (theoretically) reduce the old signal to zero (no signal left). However, because of real world effects (i.e. the shape of the hysteresis curve), some small fraction of the old signal will remain. The noise level will stay the same, though, thereby dramatically reducing the SNR for the old data. We should also remember that the write process is non-linear.

Thus we can summarize that the overwrite process has two effects: First, the SNR is reduced well below the limit that permits reading and decoding the signal, and second, whatever small fraction of signal is left has undergone a non-linear transformation, which means the linear read model is no longer valid.

This means that the resulting total signal after overwriting data is NOT the new signal plus a small, attenuated version of the old signal, but rather the new signal plus a seriously (nonlinearly) distorted residue from the old data drowned in head and media noise.

Thus far, we have been unable to find any information that would make it possible to convert this noisy residue back to the original data.

Based on this we can conclude that overwriting is a secure way of erasing data. This does require, however, that the entire physical area used to store data be truly overwritten.

Electromechanical access

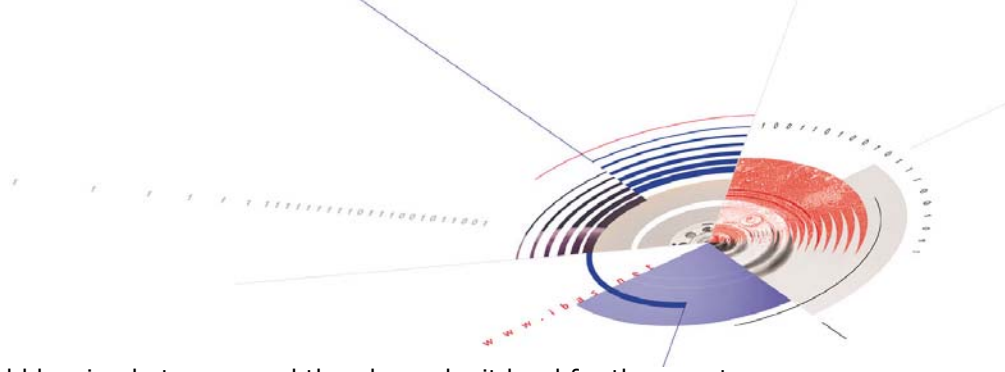
Since the different data areas on a hard drive are accessed using an electromechanical servo system, there is always the possibility of errors called TMR (Track Mis-Registration), and a small risk that portions of the old data area are not physically overwritten. Although the portions not overwritten due to TMR are very small, and extremely hard to decode, it is possible at least in theory.

The cause of TMR is mainly spindle run-out (NRR) and air turbulence; therefore the tracking errors are randomly spaced around the track. This means that multiple write passes (revolutions) can eliminate the probability that an area is not overwritten.

Thus, we can conclude that properly designed software can be used to securely erase data. The next section will discuss which properties such software should have.

3. Software Requirements

The previous section showed that overwriting using properly designed software tools can be used to securely erase data from hard drives. In this section we will discuss three properties we feel are desirable in such a tool:



1. **Easy to use:** The tool should be simple to use, and thereby make it hard for the user to make mistakes inadvertently leaving data on the drive.
2. **Complete erasure:** The tool should erase all user data on the entire physical drive.
3. **Audit Trail:** The tool should leave an audit trail such that each erasure can be logged and verified.
4. **OS Independent:** The tool must be independent of the OS that is resident on the drive where data is to be erased.
5. **BIOS and Firmware independent:** The tool should not rely on the BIOS or particular firmware of the resident machine for proper erasure.
6. **Technology independent:** The tool should run successfully on all drive technologies including SCSI and IDE.

In order to attain these properties, it is necessary to pay special attention to the following:

User Interface: The user interface should be clear and make the program easy to use. The user should be asked to make as few decisions as possible, and be presented with only the information necessary to make those decisions. This prevents 'information overload', and minimizes the possibility of overlooked or misinterpreted information.

Sanitizing: The tool should be a sanitation tool, i.e. a tool that overwrites all blocks on the drive. This means that the tool does not have to pay attention to the logical structure of the data stored on the drive, thereby eliminating the risk associated with configuration errors in the partition table, the FAT/MFT, or other system areas.

Independent of OS: An erasure tool that utilizes the OS API must relate to the logical structure on the drive. In particular the tool may use the partition table, which is used to partition a large physical drive into two-or-more smaller logical drives (typically called C: and D:). This means that the logical structure stored on the drive will influence which areas on the drive is actually overwritten. Furthermore, some tools that *do not* work independently of the OS will not handle drives formatted for all Operating Systems (An example of this is the commercially available tool *WipeInfo* which cannot be used to erase drives formatted for MAC or Unix).

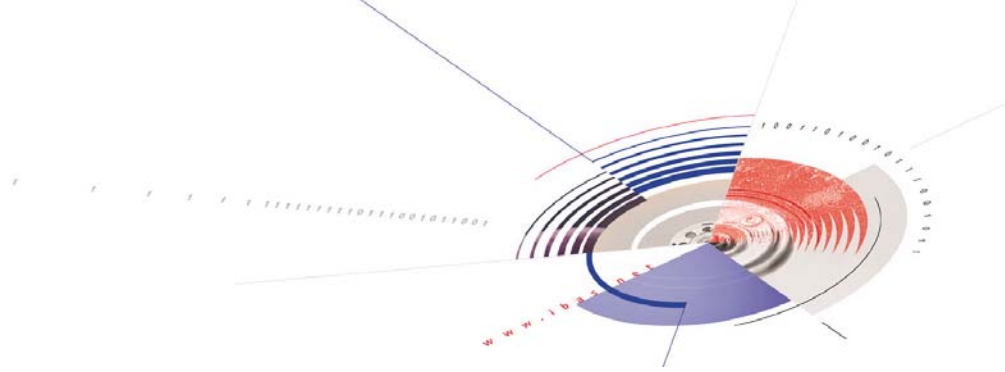
Independent of BIOS: An OS usually lets the BIOS handle low-level communication with the physical drive. However, there are many different brands and versions of BIOS in the market, and they may handle the disk drive differently. The BIOS is involved in at least three operations that could preclude secure and complete erasure:

1. Host Protected Area Feature Set:

Modern ATA drives have the capability to reserve areas of the drive. These areas can only be accessed through special commands defined in the ATA specification. Some BIOS use this capability to store system information. One example of this is the hibernation feature found on laptop computers. The BIOS reserve is an area on the drive where it stores information from open windows, applications, etc. before it powers down to preserve battery power. The BIOS does not give users or programs access to this hibernation area.

2. Geometry translation:

The blocks on a drive can be accessed by specifying a CHS address (Cylinder Head Sector). However, modern drives do not have a constant geometry across the platter surface (there are more sectors/cylinders on the outer edge of the disk than on the inner.) Therefore, the storage capacity of modern drives is specified in a number of blocks (LBA capacity). But for compatibility, the BIOS and drives still support CHS addressing. Most drives actually support different combinations of CHS parameters, also called CHS translation. When the chosen CHS translation does not 'go evenly' into the LBA capacity, we get what is referred to as orphan sectors. These are blocks on the drive that cannot be addressed with CHS addresses, but only with LBA addresses. A very simple illustration is a drive with 10 blocks, and a CHS translation C=2, H=2, S=2. With this translation it is only possible to address 8 (2x2x2) out of 10 blocks.



3. Last cylinder handling:

Some BIOS reserve certain cylinders of the drive for its own use. These cannot be overwritten using the BIOS API.

Based on this, we recommend that erasure tools not use the BIOS to access hard drives.

Handling of media defects: Even drives of high quality may develop media defects. If an erasure tool encounters soft-errors, this should be reported to the user, and the erase process should be stopped, or considered void. This kind of problem may indicate that the drive has bad-spots on its media, and that it has re-allocated blocks. Where possible (for example, SCSI drives), such re-allocated blocks should also be overwritten.

Audit Trail: The tools should provide documentation that the erasure has been completed successfully. The documentation should include all relevant information about the erasure process. Furthermore the documentation should be tamper proof, and make it possible to cross check several sources to verify the information.

4. Technical Description of ExpertEraser

In this section we will describe the erasure tool ExpertEraser developed by IBAS. We will cover two topics: The method of operation, and the quality control system.

ExpertEraser Method of Operation:

ExpertEraser uses overwriting to achieve secure erasure of data. This means that the old, possibly sensitive, data is replaced by new insensitive data. ExpertEraser is a sanitation tool, which means that all user data areas of the drive are overwritten. The ability to access data areas on a hard drive is determined by the way the drive is accessed by the software.

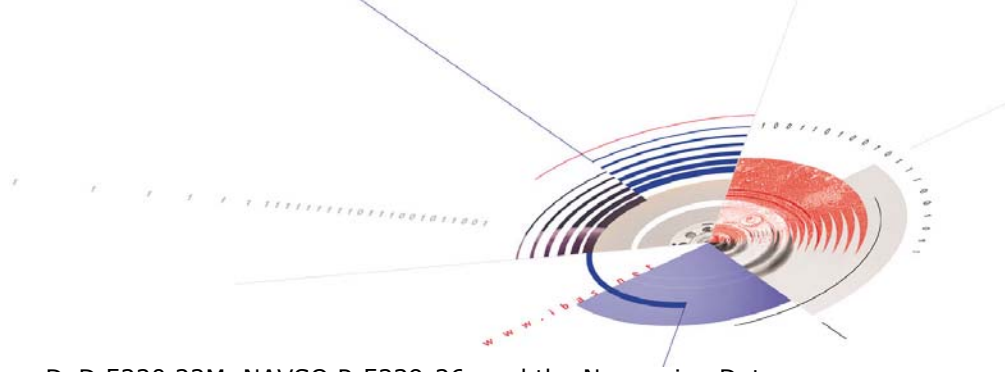
Hard drive access:

ExpertEraser can erase both SCSI and IDE type drives. Of key importance for both types is that all areas are accessed and erased. The normal way an application communicates with hard drives (SCSI and IDE) is through the BIOS API. This is fine for normal use and operation of a computer system, but it has its drawbacks when it comes to secure erasure of data. One drawback is that the application does not have direct access to the drive, and is thereby limited to the services that are made available through the BIOS API. Another problem is that the BIOS may reserve certain areas of the drive for internal use. These areas could contain sensitive information, but cannot be accessed through the BIOS Service (Int 13). Because of this ExpertEraser communicates directly with the drives that shall be erased. For SCSI drives ExpertEraser utilize the SCSI-2 standard (ANSI X3.131-1994) through an ASPI driver (Advanced SCSI Programmer's Interface). For IDE drives ExpertEraser communicates directly with the EIDE interface hardware using the ATA/ATAPI-4 specification (ANSI X3.*** 199?, Draft T13/1153D).

In having direct control, ExpertEraser can access and erase all data areas on a drive. This also means that ExpertEraser is independent of any previous partitioning of the drive, operating system and BIOS.

In summary: Because ExpertEraser communicates directly with drives, it can access and erase all data areas of a drive as required by sanitation tools.

As we have seen in the "Hard Drive technology" section, a small residue from the old data can still be detected after being overwritten. The fact that it is possible to detect (but not decode) remnants of old data has led to erasure specifications that in some cases require more than one pass for overwriting.



Three examples of such standards are DoD 5220.22M, NAVSO P-5239-26, and the Norwegian Data Security Directive (DSD):

DoD 5220.22M: Fixed Disk media should be sanitized by the following process: First write a pattern to all locations on the drive, then write its complement, and then a third pattern.

DSD: Depending on the classification and new environment of the data, the DSD require one or seven times overwriting. (See. <http://fo.mil.no/sikkerhetsstab/dsd>)

There are presently two levels of erasure available in ExpertEraser: Level 1 and level 2.

Level 1: Sanitize the media by overwriting all data locations once. The pattern written is a block full of question marks (hex code 3Fh) with a header that is used to aid the QC system built into ExpertEraser.

Level 2: Sanitize the media by overwriting all data locations seven times.

- Pass 1: Pattern is a cryptographically secure random sequence. (Produced by the ISAAC algorithm)
- Pass 2: Pattern is all zeros (00h)
- Pass 3: Pattern is all ones (FFh)
- Pass 4: Pattern is all zeros (00h)
- Pass 5: Pattern is all ones (FFh)
- Pass 6: Pattern is all zeros (00h)
- Pass 7: Pattern is "IBAS", with a header similar to the one used for level 1.

Level 2 erasure meets the requirements of the standards mentioned above.

ExpertEraser QC system:

In addition to the algorithms used to securely sanitize a drive as described above, ExpertEraser also contains a tamper proof system for quality control (QC). This system is designed to do three things: To document the erase process, to provide an audit trail (such that each erasure process can be traced), and to detect discrepancies in the technology or procedures.

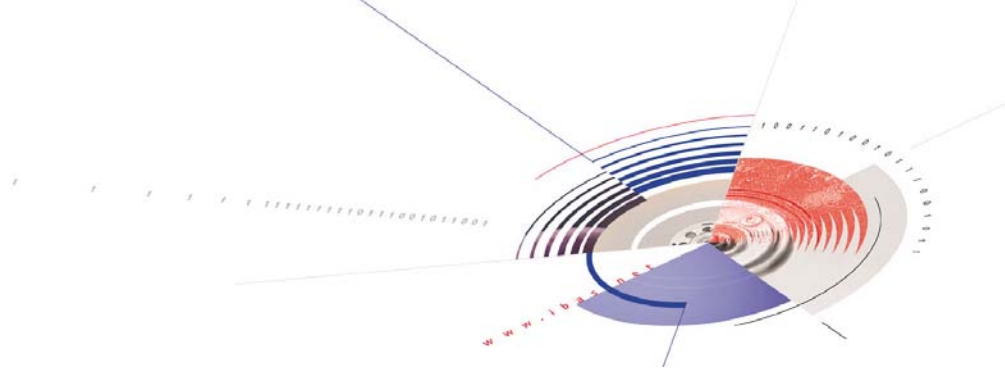
ExpertEraser achieves these goals by recording information about each erasure in several different places:

Report file: For each erasure ExpertEraser generates a report file. This file is an ASCII file that contains (at least) the following key information:

- Time and date erase process started
- Level of erasure
- Security key serial number
- Status. Specifically, how the erasure process terminated.
- Storage unit identification information (derived from serial number)
- Capacity of storage unit
- Serial Signature -- a signature derived from storage unit serial number
- Certificate Signature -- a signature to protect information that goes on the certificate.
- Report Signature -- a signature protecting vital information in the report.

Security button: To aid in license and quality control, ExpertEraser utilizes a hardware device called Dallas iButton. This device is connected to one of the PC's IO ports (printer/serial), and is a secure, password protected storage device. During the erase process, ExpertEraser records information about the erasure. When these buttons are refilled, the information stored on them will be analyzed and archived. For each erasure the following data is recorded:

- Time and date erase process started
- Level of erasure



- Capacity of storage unit
- Status. Specifically, how the erasure was terminated.
- Progress (how far has the process progressed 0-100%)
- Storage unit identification information (derived from serial number)

Storage unit: After the last overwrite pass in an erasure each block will have the following information:

- Time and date erasure process started
- Level of erasure
- Security key serial number
- Storage unit type and model number
- Storage unit serial number
- Electronic signature for the information

Erasure Certificate: When the data owner wants to document that the data is erased, he will get a certificate. To have a proper audit trail, the certificate will (at least) contain the following data:

- Time and date erasure process started
- Level of erasure
- Security key serial number
- Storage unit model number
- Storage unit serial number
- ExpertEraser version number
- Electronic signature for this information

The trace ability is achieved by cross checking these sets of information.

We also understand that it is of great importance to prevent and detect the tampering of any processed data. There are two mechanisms that will prevent and/or detect any tampering with the audit trail.

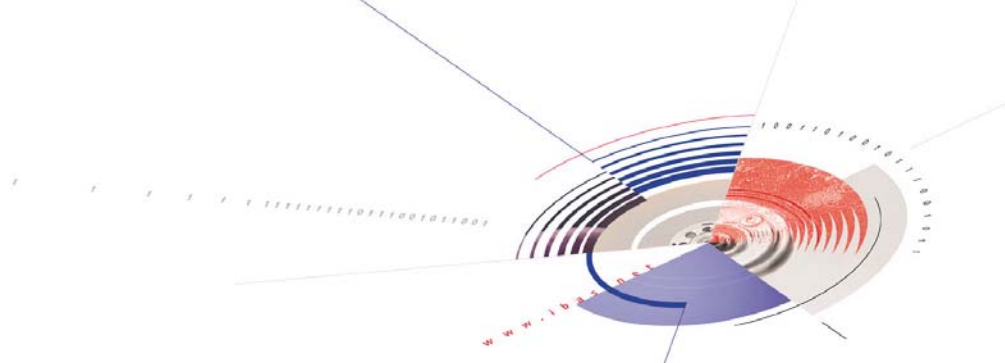
The first mechanism is the security built into the Dallas iButton itself. ExpertEraser uses the DS1991 iButton, which provides secure password protected data storage (<http://www.ibutton.com>). This means that in order to access and alter information inside the button, a 64 bit id, and a 64 bit password is required. Without these there is no way to change the information stored in the button.

The second mechanism used to detect tampering is a 128 bit electronic signature. ExpertEraser uses a secret password to seed the MD5 algorithm, which generates these signatures (for information about MD5, see <http://theory.lcs.mit.edu/~rivest/rfc1321.txt>). There are four such signatures built into the system:

Serial Signature:	This is a signature derived from the storage unit serial number
Certificate Signature:	The trace information that goes on the certificate is protected by a signature.
Report Signature:	All vital information in the erasure report is protected by a report signature.
Header signature:	The trace information written to the storage unit is protected by a signature

These two mechanisms will ensure that any tampering with the audit trail can be detected.

5. Conclusion



In this paper we have covered the topic of erasing hard drives using a software tool. We have discussed the physical and theoretical aspects of writing and reading magnetic media, and seen that if we can ensure that all areas of the physical disk are written, we will have confidence in the erasure process.

We have also seen which properties a software tool should have to ensure that 100% of the data areas are covered and erased.

Finally, we have given a technical description of ExpertEraser-- a product that was designed specifically to have these properties.