

Secure Deletion and the Effectiveness of Evidence Elimination Software

Simon Innes
School of Computer and Information Science
Edith Cowan University
nme@arach.net.au

Abstract

This paper will discuss and analyse the different methods of wiping media to make them forensically clean. This will include naming the tools, running them on a device and seeing what the device logically looks like after it has completed. It will then follow on to analyse the effectiveness of software that is designed to eliminate evidence (such as web browser history) from a computer. This analysis will take place on a small FAT32 partition running Windows 98. The test environment will be limited to using only internet explorer. The procedure will consist of installing a 'vanilla' test system, taking a bitwise copy and recording the md5. Websites will be browsed and recorded and then the system will be imaged again. After this the software will be installed and run and the 2 images will be compared. The main things that will be checked will be the temporary internet files and the registry. This will be carried out with at least 2 separate pieces of software.

Keywords

Forensic, wipe, secure delete, evidence elimination

INTRODUCTION

One of the exciting new areas that is developing in the industry of technology is the science of Computer Forensics. As Computer Forensics is such a new area, there is much discussion regarding the correct methods of implementing the tools associated with the field.. One of the topics within the Computer Forensics field involves the recovery of data from formatted hard drives and the state of the storage on which disk images are analysed. There are many tools available for use in recovering data from forensically clean devices.

The main industry standard for cleaning a device has previously been DoD 5220.22 (Deutch, 2003). This standard states that for a device to be classified as clean, it needs to be written over with three passes. The first pass consists of all 0x00, the second pass is 0xFF and the final pass is random. As time has progressed this standard has evolved into a more advanced, more secure criteria. DoD 5200.28-STD is similar to its predecessor, different in that it runs seven passes rather than three and uses much more random input. The pattern for this is to write 0x00, 0xFF, random, random, random 0xFF, 0x00 (Grinaker, n.d.)

Another method of secure deletion was a method created by Peter Gutmann in his paper 'Secure Deletion of Data from Magnetic and Solid-State Memory' (Gutmann,1996). This method indicates that eight passes must be completed on a device, with the pattern 0x00, 0x55, 0xAA, 0xFF, 0x00, 0x55, 0xAA, 0xFF. The data from this method should ideally be difficult to recover, due to it being overwritten five times. There is no random pattern generation happening, thus it is a rapid method of cleaning a device.

There is an array of tools that currently exist to help make a device forensically clean. This paper will look into several of these and outline the similarities and differences between them.

The first piece of software to be discussed is called "Wipe" (Wipe: Download, 2000). Wipe is a powerful tool as it allows the user to wipe anything from a single file to an entire hard disk. Wipe will write random data to the disk with a default of 4 passes. It is extremely configurable from command line options. It is possible to change the number of passes made and even the random device or seed used. Wipe was run on a test system and the device appearing as follows once finished:

```
00000000  ./...3h..._...9...B...ic...o.....O.E..9.....H _nx3..h..L..b..&.
00000040  O."{.../..[.K...t..nh....._}..4>.i?.^.m..$K.BQ.Yv..D...b .
00000080  .[....2.j....B..N=....|.....u.aI~za.6O..U..(.4..}..y\..A..f.f
000000C0  g/.\...+...~...\..yt.....@.K.#H.....{^...&.c}}.....Hr.....W.....
00000100  ..:6....#.F3<6..0....D.qu...Q..]a..m.<...Qd`.8.....O...j.@Pjv
```

A simple, yet somewhat effective way of doing the same thing is to simply use dd. Dd can be used in having the input file set to /dev/random and the output as the disk to be written to. As many passes as needed may be performed on the disk. Here is a sample output from a single pass using dd:

```
00000000  %.^.,...A.1.7B..9...6u...k...%.0....T\....'T0%.b..X.'..&!.....
00000040  &..Y.,...?.....=.Q>..p.X.C....6.>Is?...r.W.$....R^.....!...z..p..
00000080  D..Z...t.w.H<..0a.R.2k.yd...*R.9..#...z...Qu.I....3.p]...".....,|
000000C0  .i.Wgk.=.....`V....%\x.%Ml.>...w.w....Q%...S.O.....x..Z.3.O
00000100  ...O...`.<.....Yy.q.V`J." .8oD..wr.{...yF.}.}.}.P.:.....q1`...@
```

Reset (Grinaker, E. n.d.) is a piece of software designed to run from a floppy disk and boot a condensed linux kernel to erase a device. Reset employs several different techniques for wiping a device. In the simplest instance, Reset will set all bytes on the disk to be 0. Alternatively the aforementioned DoD 5220.22 standard or DoD 5200.28-STD may be used. The last option available in Reset employs Peter Gutmann's method of data erasing.

The final piece of software to be analysed is a more commercial option. It is called GDisk (Symantec, 2005) and was written by the company Symantec. GDisk was originally a tool that was designed for disk partition management and recovery; however it also contains a disk wiping facility. In contrast to the other software, GDisk has been designed to run in both a Windows and DOS environment. GDisk is command line based and works by erasing a disk using DoD 5200.28-STD or by using a customized number of passes.

As demonstrated, most pieces of software designed to securely erase a disk follow similar methodology and processes to achieve the desired outcome. The next section of this paper will look at tools that are designed for removing specific items from a disk as opposed to wiping the entire disk.

As technology and the internet progress, companies develop software to handle different tasks. One of the ever growing fields of software is 'evidence elimination' software. This is software that is designed to remove the traces of someone's internet browsing history. There is a range of widely available adult and illegal material on the internet, thus a market exists for this software, as web users may attempt to delete their browsing history. This paper aims to analyse the effectiveness of various evidence elimination software.

The research

Using a simple test environment, research was conducted to analyse the effectiveness of different 'evidence elimination' software. The process of the research and the outcomes will now be discussed.

The test environment

The hardware for the test environment consisted of a Pentium 200MMX with 192MB of RAM and a 4GB hard drive. The system had a floppy drive and a cd-rom and was connected to a network via an on-board network interface card. The hard drive was broken into 2 separate partitions, one 500mb partition formatted with FAT32 and a 3.5 GB partition formatted with an ext2 filesystem.

The software use in the environment consisted of Windows 98 for the purpose of executing the software. The version of internet explorer was updated to 6.0. A distribution of the Helix live cd was used to execute the analysis along with a copy of "Spider" for Windows which checks various locations for files such as index.dat.

The Contenders

For the purpose of these tests, three pieces of software were chosen. All software was either freeware or had a trial period. They are as follows:

- A. Window Washer – accessed from www.webroot.com – “Wash away all traces of your PC and Internet activity and improve system performance.”
- B. History Kill 2003 – accessed from www.historykill.com – “The #1 rated privacy tool on the internet!”
- C. History Swatter – accessed from www.historyswatter.com – “Free Internet History Eraser!”

This software was located through various searches on the search engine google.com.

The Process

To start with, a fresh installation of Windows 98 was made on the 500mb partition (we will call it hda1). Internet Explorer 6 was then installed. Once the installation was completed, several web pages were loaded via internet explorer and noted so they would be easy to find. Among these sites was a site of an adult nature that spawned a many popups which potentially left a lot of data remaining. After the websites had been browsed, the system was restarted and Helix was loaded. The 2nd disk partition was then created with an ext2 file system (hda2):

```
[Helix (mnt)]# mkfs.ext2 /dev/hda2
mke2fs 1.35 (28-Feb -2004)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
405600 inodes, 810432 blocks
40521 blocks (5.00%) reserved for the super user
First data block=0
25 block groups
32768 blocks per group, 32768 fragments per group
16224 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912
```

```
Writing inode tables: done
Writing superblocks and filesystem accounting information: done
```

```
This filesystem will be automatically checked every 39 mounts or
180 days, whichever comes first. Use tune2fs -c or -i to override.
```

‘Wipe’ was then run on the file system with seven passes to ensure that it was forensically clean.

Once the device was clean, the windows partition was imaged. This was done using dd with the 500 MB windows drive as the source and the file ‘first-image’ on the 2nd partition being the destination:

```
dd if=/dev/hda1 of=./first-image;
1056321+0 records in
1056321+0 records out
540836352 bytes transferred in 469.378755 seconds (1152239 bytes/sec)
```

Once this was done, hash checks were made to ensure that the copy was identical to the drive itself:

```
MD5
caa25d3bd04b61941646073a5a568388 /dev/hda1
caa25d3bd04b61941646073a5a568388 /mnt/hda2/first-image
```

```
SHA1
e4764646153e81f1e7b8074f58dd441d30b77435 /dev/hda1
e4764646153e81f1e7b8074f58dd441d30b77435 /mnt/hda2/first-image
```

```
CRC32
2867881248 540836352 /dev/hda1
2867881248 540836352 /mnt/hda2/first-image
```

After backing up the original image, the first piece of software was tested.

Contender A: Window Washer

The first piece of software used was ‘Window Washer’. The most interesting thing about Window Washer was that it was advertised as having a ‘Bleach’ function that overwrites files with random characters and makes them

unrecoverable and undetectable to unerase software. Apart from that, items were simply removed from the recycle bin, registry, windows temp files, index.dat, recently opened documents and recently viewed pictures.



Figure 1 – A summary of the tasks that Window Washer has carried out.

Window Washer was installed and run without difficulties. All the things that were ‘cleaned’ are visible on the above screenshot (figure 1). Upon completion of this, Spider was run to investigate the various index.dat files that windows creates. The output was as follows:

```
{\rtf1\ansi\deff0\defstab720{\fonttbl{\f0\fswiss MS Sans Serif;}{\f1\froman\fcharset2 Symbol;}{\f2\froman Times New Roman;}{\f3\froman Times New Roman;}}
{\color\red0\green0\blue0;}
\deflang1033\pard\plain\fs18 Spider Log File - Copyright (C) 1999 - Ward van Wanrooij <ward@ward.nu>
\par
=====
=====
\par Scanned c:
\par
\par Files Scanned:
\par c:\\WINDOWS\\Temporary Internet Files\\Content.IE5\\index.dat
\par c:\\WINDOWS\\Cookies\\index.dat
\par c:\\WINDOWS\\History\\History.IE5\\index.dat
\par
=====
=====
\par URLs Found:
\par ***** Scanning c:\\WINDOWS\\Temporary Internet Files\\Content.IE5\\index.dat... *****
\par
\par http://www.microsoft.com/isapi/redir.dll?prd=ie&clid=0x0409&pver=6.0&ar=ienews&os=98
```

```

\par http://v4.windowsupdate.microsoft.com/
\par
\par ***** Scanning c:\\WINDOWS\\Cookies\\index.dat... *****
\par
\par
\par ***** Scanning c:\\WINDOWS\\History\\History.IE5\\index.dat... *****
\par
\par
\par
=====
=====
\par
\par }

```

After this, the registry was also searched to ensure that nothing could be found. This assumption was accurate.

After the software had been run, Helix was booted and the analysis method took place. The image was made using dd and the hashes were recorded.

MD5

```

e7608fb190deb99be92b2ad4849e6e38 /dev/hda1
e7608fb190deb99be92b2ad4849e6e38 /mnt/hda2/first-image-after-ww

```

SHA1

```

48e4f721e55dc4f1ee01fd2e7916098644395ebf /dev/hda1
48e4f721e55dc4f1ee01fd2e7916098644395ebf /mnt/hda2/first-image-after-ww

```

CRC32

```

922612215 540836352 /dev/hda1
922612215 540836352 /mnt/hda2/first-image-after-ww

```

Because the websites that had been browsed with this computer had been recorded, hexedit was used to search through the image. Just searching for the names of the websites was enough to uncover some data relating to them.

The first batch of data that was found appeared to be hidden in the windows registry. Interestingly, there was no trace of it through regedit in Windows.

Note: The hexadecimal values have been removed as they irrelevant.

```

1033A4D4 ...RegBackup].....].0.....,
1033A500 .3bdd6b017b35029e,Software\Microsoft\Interne
1033A52C t Explorer\Main,.w.....w.....0.map..
1033A558 .....B.3bdd6b017b35029e,1,HKCU,Software\M
1033A584 icrosoft\Internet Explorer\Main,First Home P
1033A5B0 age,w...}...w.....1001.....Desc
1033A5DC riptionInternet Explorer.....FileName
1033A608 IEXPLORE.EXE.....Version...v.....v
1033A634 .....{89820200-ECBD-11cf-8B85-00AA005B
1033A660 4383}.Restore.....Version6,0,2800,110
1033A68C 6.....Localeenj.....j...&.....{9E
1033A6B8 F0045A-CDD9-438e-95E6-02B9AFEC8E11}.....
1033A6E4 ...Version1,0,2195,0.....Locale*"...
1033A710 ...".Help_Menu_URLs.....

```

```

1033A73C .....Discardable.....PostSetu
1033A768 p(.....(.....Component Categories:...
1033A794 .....:...&.....{00021493-0000-0000-C000-00
1033A7C0 0000000046}o.....Q.....Toolbar.....
1033A7EC .....LinksFolderNameLinks....9-5E....Locked
1033A818 ....rder.....9-5E....Locked.....%.
1033A844 .....RunMRU.....awinipcfg\1.....
1033A870 .....MRUListacb......b\192.168.0.3\
1033A89C Public Drop Zone\1.....s.7....cregredit\1....
1033A8C8 .....Count.....HRZR_PGYFRF
1033A8F4 FVBA.v%.....3\P....HRZR_PGYPHNPbhag:pgb
1033A920 e.....HRZR_HVGBBYONE.
1033A94C .....@R.R.....HRZR_HVGBBYONE:0k
1033A978 1,125.....@R.R.....x.....Typ
1033A9A4 edURLs.....url1http://www.pornmovies.
1033A9D0 com/.....url2http://www.hotmail.com/.
1033A9FC .....url3http://www.sexpics.com/.....
1033AA28 .....url4http://www.msn.com.....
1033AA54 ....RunMRU.....awinipcfg\1.....
1033AA80 .MRUListgfdacb......b\192.168.0.3\Pu
1033AAAC blic Drop Zone\1.....s.7....cregredit\1...mr
1033AAD8 5.....d\192.168.0.3\c$\1.....e\192.
1033AB04 168.0.3\d$\1.....f\192.168.0.101\1.

```

The labels in this part of the disk image look like this;

```
{89820200-ECBD-11cf-8B85-00AA005B 4383}
```

```
Software\Microsoft\Internet Explorer\Main
```

This can be interpreted to mean that this information is stored somewhere in the registry. As can be seen, the URLs that exist here have the label url1, url2 etc. in front of them, which would lead one to believe they are to do with the visited websites.

Another interesting find after running Window Washer was this;

```

0D780440 .....
0D78046C .....REDR.....Q...tG.http://w
0D780498 ww.pornmovies.com/.....
0D7804C4 .....

```

There were several of these 'REDR' tags containing previously browsed URLs. Interestingly, there is nothing else surrounding them but what appear to be blank areas on the image.

The next major find in this analysis was the actual presence of cookies. It is difficult to say where exactly in a Windows environment these existed because doing a regular search for them did not return any results.

```

0E1C00E4 000&lc=1033&_lang=EN.....URL .....*.....
0E1C0110 .E.....Q..`...h.....
0E1C013C .....J1.q.....
0E1C0168 :2004101020041011: Simon@:Host: loginnet.pas

```


'Foremost' is a program which has the ability to retrieve files from a raw disk image. After locating the actual html code within the disk image, foremost was tun to investigate whether it was possible to recover any files. Foremost took roughly ten minutes to analyse the image, in that time several html pages that had been browsed and allegedly deleted were located.

Upon completion of the analysis of Webroot's Window Washer, it would seem that the tool deletes all history and keeps it well hidden from a standard Windows environment. This would prove useful if the user examining the system only searched for the data in Windows, however a more thorough forensic investigation would reveal remaining evidence. It can be concluded that due to the recovery of HTML on the disk, this product is second rate and would not be value for money if purchased.

Contender B: History Kill 2003

The second piece of software used in this research was 'History Kill 2003'. History Kill has an interesting feature called a 'File Shredder' which, according to the website from which the software is available, will:

"Encrypt and overwrite your web surfing tracks 21 times (or more) so that no one can undelete or recover your web tracks! HistoryKill defeats forensic software used by the US Secret Service, Customs Department and LAPD!" (History Kill, 2005)

The amount of times History Kill overwrites this data is configurable before running. For the purpose of this exercise, twenty one, the default, was chosen.



Figure 2 – HistoryKill's options for Internet Explorer and MSN Privacy



Figure 3 - HistoryKill's options for Windows Privacy.

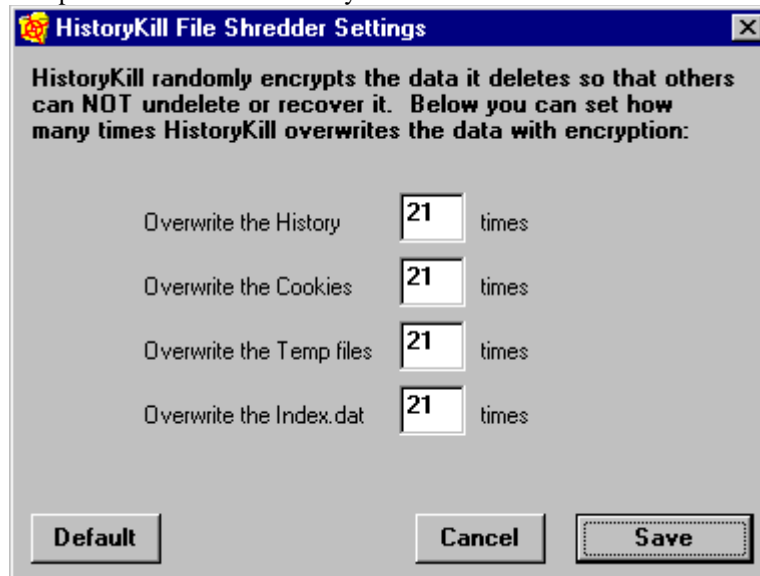


Figure 4 - HistoryKill's options for "file shredding".

As you can see from the above screenshots, (figures 2 – 4) History Kill removes quite a range of data from different locations. You can also specify how many times to 'shred' certain data.

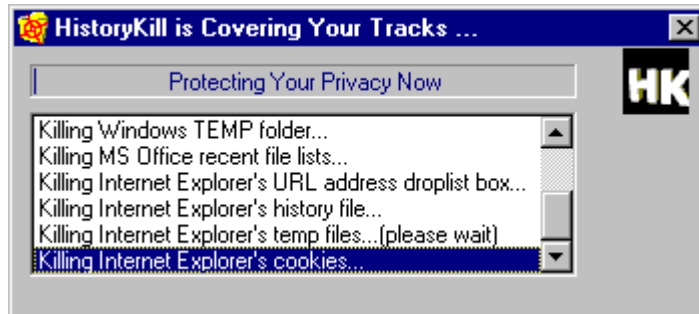


Figure 5 – HistoryKill’s progress during an execution.



Figure 6 – HistoryKill after the execution has completed.

After running the program History Kill, Spider was again executed to examine what remained on the system.

```

\deflang1033\pard\plain\f2\fs18 Spider Log File - Copyright (C) 1999 - Ward van Wanrooij <ward@ward.nu>
\par
\par Scanned C:\\WINDOWS
\par
\par Files Scanned:
\par C:\\WINDOWS\\Temporary Internet Files\\Content.IE5\\index.dat
\par C:\\WINDOWS\\Cookies\\index.dat
\par C:\\WINDOWS\\History\\History.IE5\\index.dat
\par
\par URLs Found:
\par ***** Scanning C:\\WINDOWS\\Temporary Internet Files\\Content.IE5\\index.dat... *****
\par
\par http://www.imdb.com/
\par http://www.imdb.com/
\par http://ia.imdb.com/media/imdb/01/I/21/62/48.jpg
\par http://ia.imdb.com/media/imdb/01/I/31/62/48.jpg
\par http://ia.imdb.com/media/imdb/01/I/94/09/38.gif
\par http://i.imdb.com/sok.css
\par http://ia.imdb.com/media/imdb/01/I/41/62/48.jpg
\par http://i.imdb.com/imdb.css
\par
\par
\par }
\par ***** Scanning C:\\WINDOWS\\Cookies\\index.dat... *****
\par
\par Cookie:simon@msn.com/
\par Cookie:simon@microsoft.com/
\par Cookie:simon@www.ultravideos.com/s6/
\par Cookie:simon@imdb.com/
\par Cookie:simon@maxserving.com/
\par Cookie:simon@atdmt.com/
\par Cookie:simon@www.imdb.com/
\par Cookie:simon@advertising.com/
\par Cookie:simon@servedby.advertising.com/
\par Cookie:simon@ultravideos.com/

```

```

\par Cookie:simon@pornoground.com/
\par
\par ***** Scanning C:\WINDOWS\History\History.IE5\index.dat... *****
\par
\par http://www.pornmovies.com.20041010.nudity.com/main.php
\par http://www.msn.com
\par http://www.ultravideos.com/s6/video002.html
\par http://www.pornmovies.com.20041010.nudity.com
\par
\par =====
\par
\par }

```

The output from Spider has been truncated as it located all sites that had been browsed on this machine. History Kill appears to have missed one important factor when deleting these files, as index.dat files have been located in the 3 areas that Spider checks. Another point to note is that there are cookies listed here for websites that were never browsed. It would appear that if one site is visited which is 'linked' to another, the cookies will be created.

Although the use of Spider proved that History Kill doesn't remove everything, a forensic analysis was still carried out on the device to see if anything unusual or interesting happened. The hashes for this disk image look like this:

```

MD5
247caafd62369d427018e042013173ff /dev/hda1
247caafd62369d427018e042013173ff ./after-hk2k3

SHA1
626d67ef331a09f0ec0f81d2eafdb49caf2919e2 /dev/hda1
626d67ef331a09f0ec0f81d2eafdb49caf2919e2 ./after-hk2k3

CRC32
3899082824 540836352 /dev/hda1
3899082824 540836352 ./after-hk2k3

```

History Kill appeared to leave all the same tracks that Window Washer did. There were URLs found that started with the REDR tag;

```

0D780458 .....
0D780480 REDR.....Q...tG.http://www.pornmovies.co
0D7804A8 m/.....
0D7804D0 .....

```

More cookies were found;

```

0E1C0454 .....:2004101020041011: S
0E1C047C imon@http://www.pornmovies.com.20041010.
0E1C04A4 nudity.com.....

```

And traces of HTML code were located;

```

0E316BF0 U....)E..{..t:..{<HTML>.<HEAD>.<TITLE>:::
0E316C18 PornMovies.com :::Ultra Videos!!!</TIT
0E316C40 LE>.<meta NAME="description" CONTENT="Of
0E316C68 6fers porn movies, live porn, sex picture
0E316C90 s, movies, and more.">.<meta NAME="keywo
0E316CB8 rds" CONTENT="sex,porn,porno,adult video
0E316CE0 s,xxx,porn movies,xxx videos,adult movie

```

0E316D08 s">.<style>.a:hover {color:#FF0000}.TD{f

Diagnosis: Whilst History Kill appears to delete internet history, it leaves remnants of the history to be found, This software would be recommended for amateur use, such as in a domestic setting, but would be ineffective in a professional environment where a user doing the search would know to carry out a more thorough analysis,

Contender C: History Swatter

The third piece of software to be analysed is called History Swatter. History Swatter was created by a company called Fun Web Products, and was accessible from the web, presented as an advert in a popup window. Within the content of the website, History Swatter claims to remove the following items:

Document History

Temporary Files

Disk Temp Files

Clipboard

Memory Dump File

MS Download Temp Folder

Run History

Start Menu Click History

Start Menu Order History

Page (Swap) File

Address Bar History

AutoComplete

Index.dat

Media Bar History

Temporary Internet Files

History (Visited Sites)

Cookies

Unlike the initial two tested programs, History Swatter does not boast any special deletion techniques. The only feature unique to History Swatter is that it includes Fun Web Products' own Internet Explorer Toolbar. This is a feature indicative of low grade software.

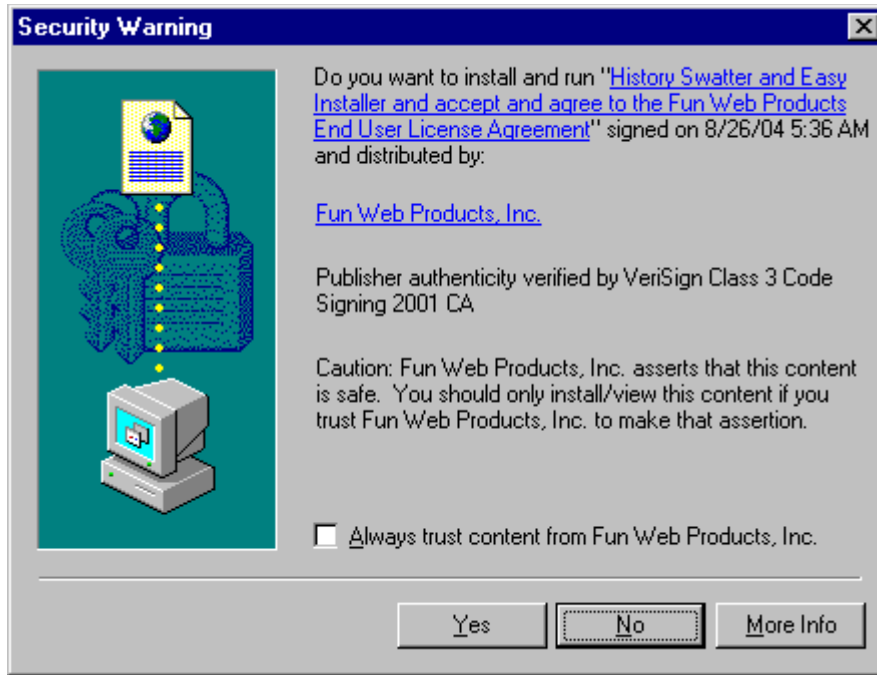


Figure 7 – The install process for History Swatter.

Rather than having to download an executable, History Swatter installs via Internet Explorer. (Figure 7) One of the reasons this software was chosen was because of the unique install method.

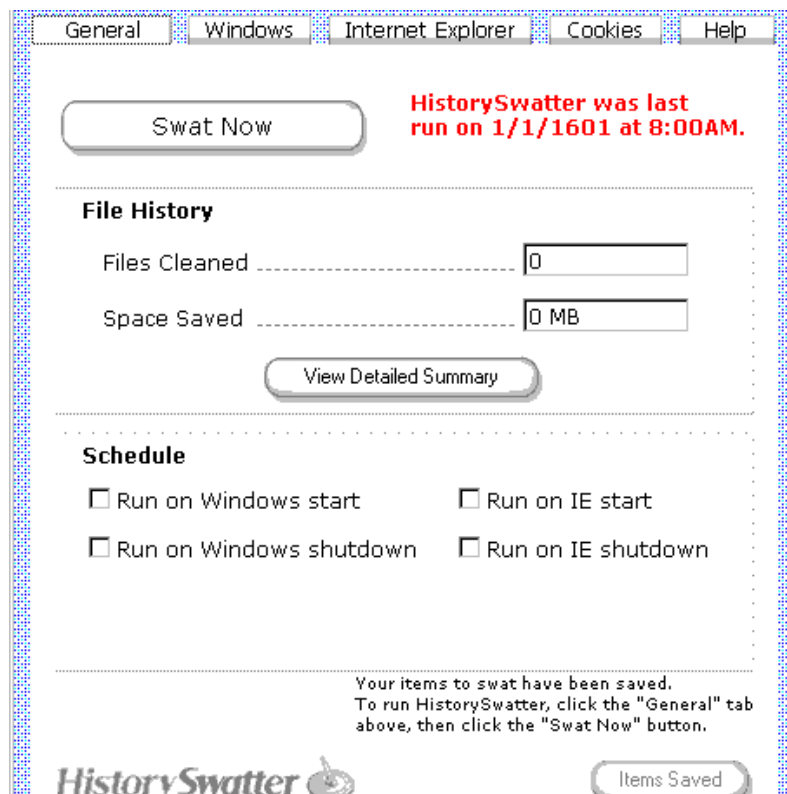


Figure 8 – History Swatter's main screen.

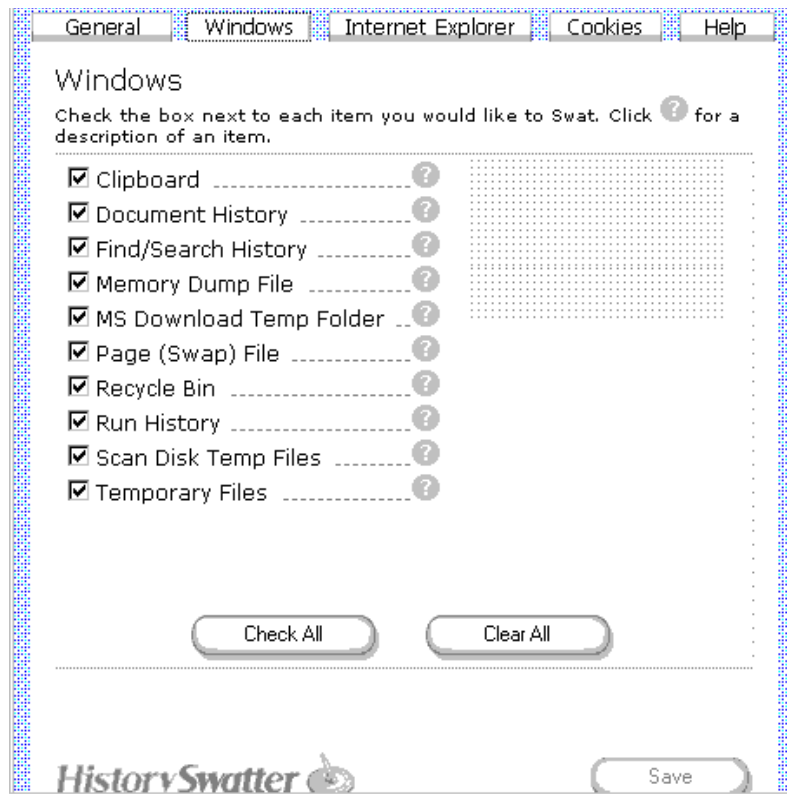


Figure 9 – History Swatter’s options for Windows privacy

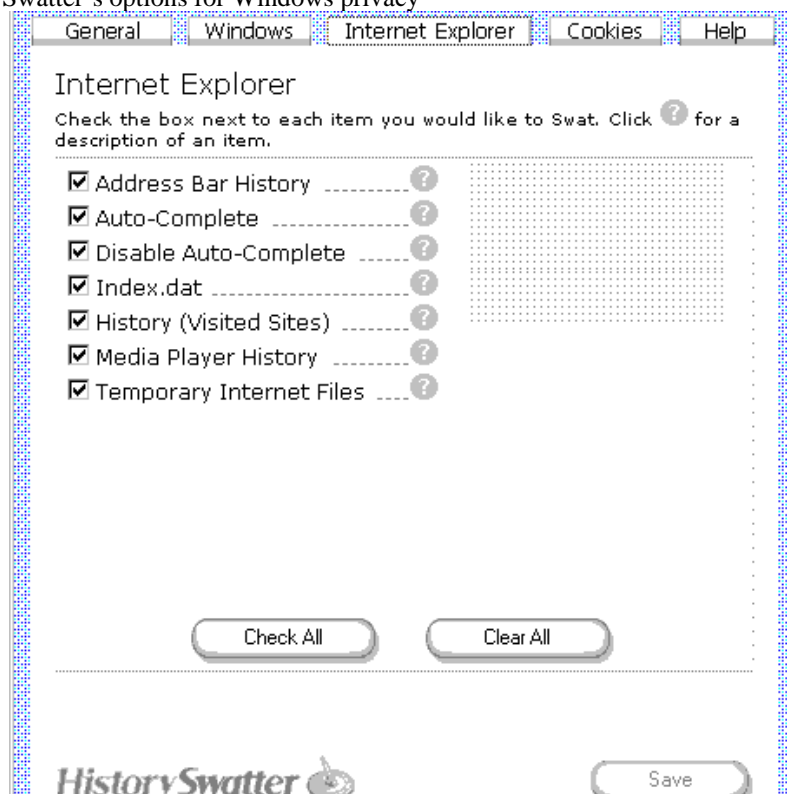


Figure 10 – History Swatter’s options for Internet Explorer Privacy

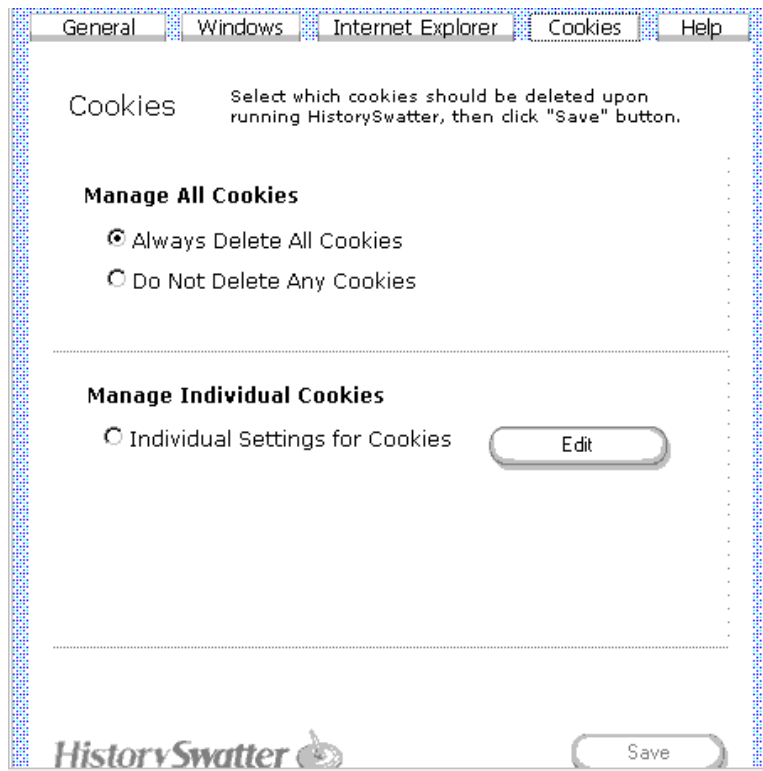


Figure 11 – History Swatter’s options for cookies.

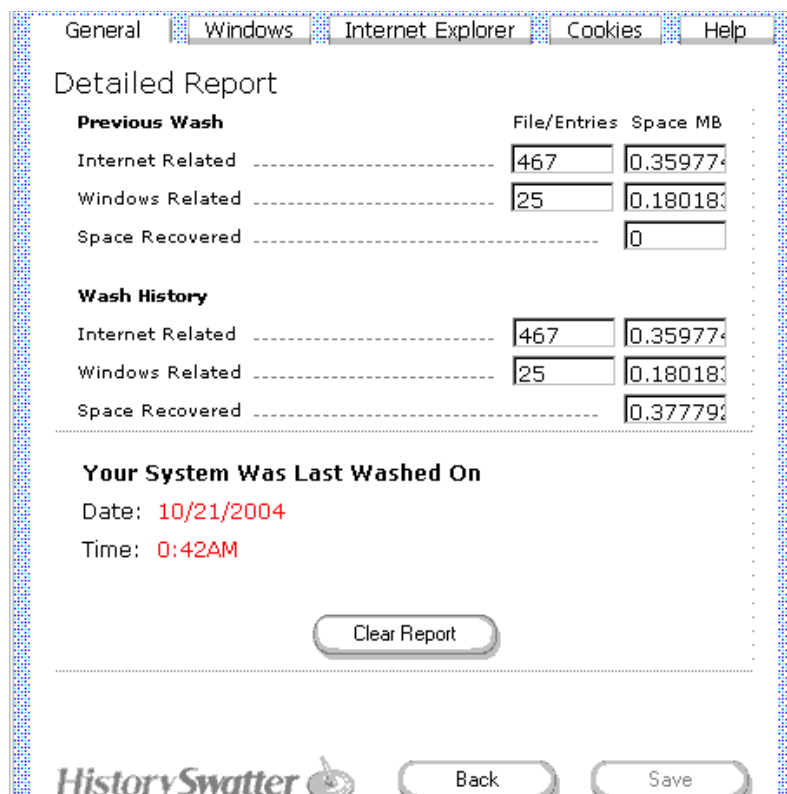


Figure 12 – History Swatters main screen after a run has completed.

In comparing the numbers here to that of the tests run on Window Washer, there are many similarities to be seen.

As has been performed with the previous analytic tests, Spider will be used to determine the basic effectiveness of the software.

```
\par =====
\par Scanned C:\WINDOWS
\par
\par Files Scanned:
\par C:\WINDOWS\Temporary Internet Files\Content.IE5\index.dat
\par C:\WINDOWS\Cookies\index.dat
\par C:\WINDOWS\History\History.IE5\index.dat
\par =====
\par URLs Found:
\par ***** Scanning C:\WINDOWS\Temporary Internet Files\Content.IE5\index.dat... *****
\par
\par
\par
\par http://view.atdmt.com/AVE/view/msnkonm00200074ave/direct;wi.309;hi.60/01/
\par http://www.imdb.com/
\par
\par http://www.imdb.com/Sections/Gallery/
\par http://www.imdb.com/Sections/Gallery/
\par http://fpdownload.macromedia.com/get/shockwave/cabs/flash/swflash.cab
\par http://www.imdb.com/rg/SECGAL/PTF//name/nm1663927
\par http://www.imdb.com/name/nm1663927/
\par http://www.imdb.com/google/box?num=3;k=power100-
withsc;placement=midbucket;rnd=96922;sid=8483;referer=%2Fname%2Fnm1663927%2F;slot=GOOGLE
\par http://www.imdb.com/name/nm1663927/board/threads/
\par http://www.pornmovies.com/
\par http://www.ultravideos.com/?revid=30032&s=6&nopop=1
\par http://www.ultravideos.com/s6/images/blonde.jpg
\par http://216.158.129.34/ml/wPCmavdMlpTfNQDA4xMA/UV/vid002.wmv
http://download.historyswatter.com/frame.jsp?partner=ZHXXXXXXXXXAU&product=historyswatter&w=h&opcr
eativeid=&partner=&opcreativeland=505%2C0%2C5494%2Csh.11.3pj
\par
\par ***** Scanning C:\WINDOWS\Cookies\index.dat... *****
\par
\par Cookie:simon@www.ultravideos.com/s6/
\par Cookie:simon@imdb.com/
\par Cookie:simon@maxserving.com/
\par Cookie:simon@atdmt.com/
\par Cookie:simon@www.imdb.com/
\par Cookie:simon@advertising.com/
\par Cookie:simon@servedby.advertising.com/
\par Cookie:simon@ultravideos.com/
\par
\par ***** Scanning C:\WINDOWS\History\History.IE5\index.dat... *****
\par
\par =====
\par
\par }
```

From this output, the conclusion can be made that History Swatter only deleted one index.dat file. It is also interesting to note that there is an entry for a .wmv file that did not appear in the output for History Kill.

In carrying out a forensic analysis, Helix was loaded, the disk was imaged and the hashes were made to verify it.

MD5

```
ee1bb1d107c645860fb77922b751c1e8 /dev/hda1
ee1bb1d107c645860fb77922b751c1e8 .after-hs
```

SHA1

```
9e4f932361fede51ce9a784123774bb76982c8a9 /dev/hda1
9e4f932361fede51ce9a784123774bb76982c8a9 after-hs
```

CRC32

3357363561 540836352 /dev/hda1

3357363561 540836352 after-hs

The analysis of this disk was very similar to the analysis after History Kill was run. Traces of cookies were located, but this time there appeared to be several more than before.

0E1A8168 :2004100420041011: Simon@:Host: www.pornmovies.c
0E1A8198 om.20041010.nudity.com.....

0E1C0468 :2004101020041011: Simon@http://www.pornmovies.c
0E1C0498 om.20041010.nudity.com.....

0E1A8858:2004100420041011: Simon@http://
0E1A8888 www.ultravideos.com/s6/video002.html.....

The mysterious REDR tag showed up again:

0D780470REDR.....Q...tG.http://www.pornm
0D7804A0 ovies.com/.....

Also more HTML code was recovered, however as it was at the same location as the previous analyses, they will not be displayed here.

There was some data in this analysis that made it different from the other two. Although it is difficult to deduce exactly then nature of this data or its origin, it may be a derivative of a script of some description.

0E462BF8revid.30032.ultravideos.com/.1536.283255
0E462C28 6928.29685132.2740218304.29667026.*.ssite.6.ultr
0E462C58 avideos.com/.1536.2832556928.29685132.2740218304
0E462C88 .29667026.*.nopop.1.ultravideos.com/.1536.283255
0E462CB8 6928.29685132.2740718304.29667026.*.refer.http%3
0E462CE8 A%2F%2Fwww.pornmovies.com.20041010.nudity.com.ul
0E462D18 travideos.com/.1536.2832556928.29685132.27407183
0E462D48 04.29667026.*.sessid.25f081fc65f17457c9ae3e5f32c
0E462D78 ffcd9.ultravideos.com/.1536.2832556928.29685132.
0E462DA8 2740718304.29667026.*.tstamp.1097417795.ultravid
0E462DD8 eos.com/.1536.2832556928.29685132.2740718304.296
0E462E08 67026.*.....

The next section, which was located not too much further into the image, is obviously java script that must have come from one of the web pages that caused the popup windows to appear.

0E467248 nopoping = new Image();...nopoping.src = "/?act
0E467278 ion=nopoping&revid=30032&nopop=1&refer=http%3A%2
0E4672A8 F%2Fwww.pornmovies.com.20041010.nudity.com";...}
0E4672D8 }.self.focus();.stayUnder();.attachEvent('onbef
0E467308 oreunload',goBUL);.../*]]> *//-->.</script>...<
0E467338 script language="JavaScript" type="text/JavaScri
0E467368 pt">.<!--.function MM_openBrWindow(theURL,winNam
0E467398 e,features) { //v2.0. window.open(theURL,winNam
0E4673C8 e,features);.} //-->.</s

The next block of data is similar to an earlier block, however this particular one appears to convey itself as being generated as a result of a php script. It would appear that all the labels and numbers contained within this block are actually php variables. It also makes reference to a .dll file.

```

102F3BE8 .dll..%.CreateBitmap..T.revid.30032.teensforcash
102F3C18 .com/.1536.3552556928.29685132.3461318304.296670
102F3C48 26.*.ssite.2.teensforcash.com/.1536.3552556928.2
102F3C78 9685132.3461318304.29667026.*.nopop.1.teensforca
102F3CA8 sh.com/.1536.3552556928.29685132.3461318304.2966
102F3CD8 7026.*.refer.http%3A%2F%2Fwww.pornmovies.com.200
102F3D08 41010.nudity.com%2Fmain.php.teensforcash.com/.15
102F3D38 36.3552556928.29685132.3461318304.29667026.*.ses

```

The final discovery found in this disk image that made it different from the previous analyses was a block of data that appeared to be a variety of history. There is a reference to 'TypedURLs' followed by a list of the URL's that had been typed in, rather than all the URLs that were linked to by another page. It appears that this data is actually from some internet explorer settings. This is evidenced by the portions of data which explain can be derived from the parts that express 'Cache_Update_Frenquency' and 'Save_Session_History'.

```

1033A640 e.lnk.....MRUListba....M.(...#.bhs-1.
1033A670 bmp...0.....hs-1.lnk.....x.....
1033A6A0 ..TypedURLs.....u
1033A6B8 rllhttp://www.historyswatter.com/.....url
1033A6E8 2http://www.pornmovies.com/.....url3http:
1033A718 //www.hotmail.com/.....url4http://www.sex
1033A748 pics.com/.....url5http://www.msn.com2...s
1033A778 ...2.....Main.....Anchor Underliney
1033A7A8 es.....Cache_Update_FrequencyOnce_Per_Ses
1033A7D8 sion.....Display Inline Imagesyes.....
1033A808 ....Do404Search.....Local PageC:\WIND
1033A838 OWS\SYSTEM\blank.htm.....Save_Session_His

```

History Swatter was a second rate software program, similar to those previously tested in that it sufficiently cleans the system for amateur use. However any further forensic testing would reveal the user's history on the machine.

CONCLUSION

In conclusion, the freeware and shareware 'evidence elimination' software that claims to offer a comprehensive elimination system fails to deliver on this promise. This evidences the expertise and extensive expertise needed to remove all traces of web browsing history. The most promising of the three pieces of software tested was Window Washer, which successfully removed all visible traces of browser history from a windows environment. Both History Swatter and History Kill overlooked some areas from which to remove data.

The software is marketed towards home or domestic users wanting to delete browser history for fear of virus attack or discovery by other users of personal browsing history. For this purpose, the software is satisfactory and will meet the needs of the user. It is not designed, however, to protect users within a business which has a forensic capability setup. In this respect, the marketing claims of the software can be misleading as it claims to have removed items, while in reality it has only removed them from sight of the naked eye. Upon conclusion of this analysis, it would seem the best way to cover the tracks of a user would be to not leave any in the first instance.

REFERENCES

- Deutch, J. M. (2003). *DoD 5220.22-M National Industrial Security Program Operating Manual* Retrieved 12 October, 2004 from <http://www.dss.mil/isec/nispom.htm>
- Grinaker, E. (n. d.) *Reset 0.1.0*. Retrieved 10 October, 2004 from <http://oss.codepoet.no/reset/README>
- Guntmann, P. (1996), *Secure Deletion of Data from Magnetic and Solid-State Memory*. Retrieved 12 October, 2004 from http://www.cs.auckland.ac.nz/~pgut001/pubs/secure_del.html
- History Kill (2005). *Features*. Retrieved 8 October, 2004 from <http://www.historykill.com/features.asp>
- Laliberte, S & Gupta, A. (2004). *The Role of Computer Forensics in Stopping Executive Fraud*. Retrieved 10 October, 2004 from <http://www.awprofessional.com/articles/article.asp?p=336258&seqNum=2>
- McKemmish, R. (1999). *What is Forensic Computing?*. Retrieved 6 October, 2004 from <http://www.aic.gov.au/publications/tandi/ti118.pdf>

- O'Conner, T. (2004). *Syllabus for Computer Forensics*. Retrieved 6 October, 2004 from <http://faculty.ncwc.edu/toconnor/426/default.htm>
- Spector, L. (2003). *Answer Line: Wipe Your Drive Clean of All Its Sensitive Data*. Retrieved 6 October, 2004 from <http://www.pcworld.com/howto/article/0,aid,110338,00.asp>
- Symantec, (2005). *Switches: GDisk*. Retrieved 8 October, 2004 from <http://service1.symantec.com/SUPPORT/ghost.nsf/docid/2000030715304425>
- Vier, T. (2004). *Wipe*. Retrieved 25 October, 2004 from <http://wipe.sourceforge.net/>

COPYRIGHT

Simon Innes ©2005. The author/s assign the School of Computer and Information Science (SCIS) & Edith Cowan University a non-exclusive license to use this document for personal use provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive license to SCIS & ECU to publish this document in full in the Conference Proceedings. Such documents may be published on the World Wide Web, CD-ROM, in printed form, and on mirror sites on the World Wide Web. Any other usage is prohibited without the express permission of the authors.