Global Information Assurance Certification Paper

Part 1: Analyze an Unknown Binary

Overview of Investigation

John Price has been suspended from his place of employment because an audit revealed that company resources might be in use for illegal transfer of copyrighted material. Mr. Price wiped the hard disk of his company PC before an investigation could be undertaken, but a single floppy was seized. Mr. Price disclaims ownership of the floppy, but a forensic analysis of the floppy makes this claim unlikely. Of particular interest is a single binary file on the floppy, named "prog". **Forensics analysis reveals that this program is a Linux executable and is a version of the data hiding tool "bmap" (see Section Program Identification for background)**. The program allows storage of arbitrary data in the slack space of a file.

An investigation of the floppy is detailed below. The following equipment and software was used in the analysis of the floppy:

Primary forensics machine:

T40p Thinkpad with (1) 60GB 7200 rpm disk and 2GB RAM running Windows XP. Forensics software used: Accessdata's Forensics Toolkit ("FTK") v1.43a.

Secondary forensics machine, used to execute the unknown binary in a controlled environment:

A31p Thinkpad with (1) 60GB 7200 rpm disk and 512MB RAM running Redhat Linux 9 with a 2.4.20-8 kernel. GCC version is 3.2.2. Filesystem is ext3.

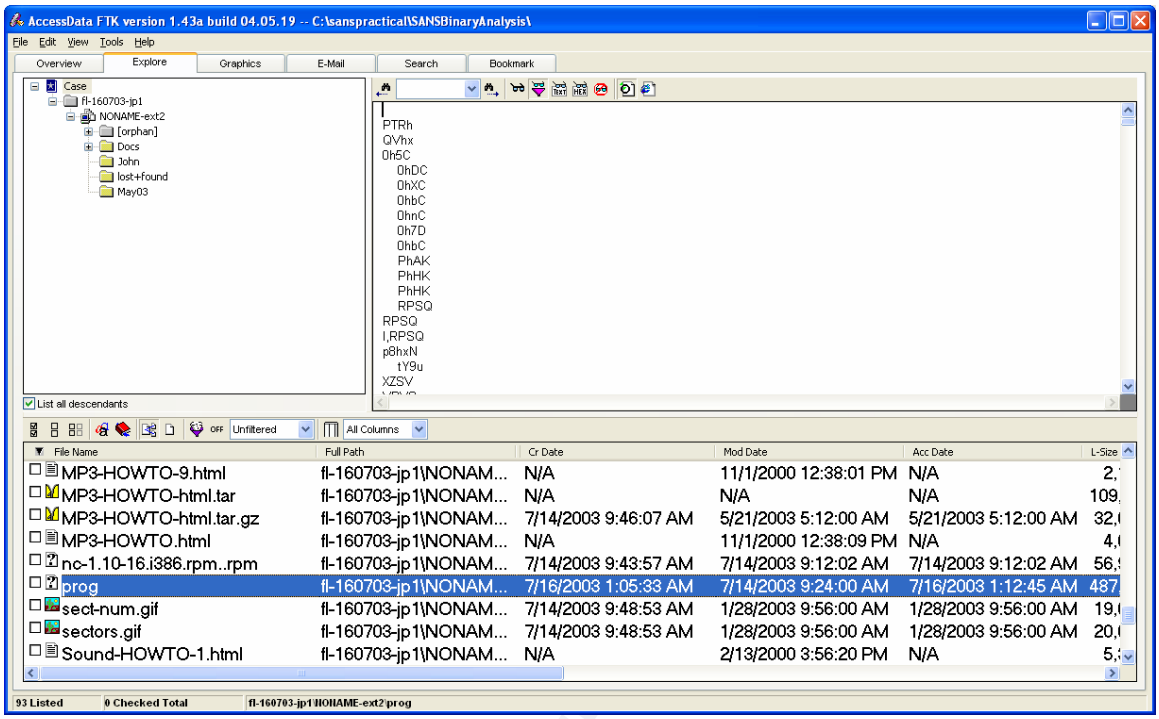
Binary Details and Program Description

The floppy image was loaded into FTK by creating a new case. Details of the FTK imaging process, options available when creating a new case, and options that control FTK's initial preprocessing for evidentiary items are covered in detail in Section "Media Analysis of the System" in Part (2) of this practical. All of the default options were used when opening the case for analysis of the unknown binary. The case name chosen was "SANSBinaryAnalysis".

FTK provided the following information on the binary "prog". This information was obtained from the Explore view in FTK after clicking on "prog". A screenshot illustrates the information.

Inode changed (file "created") 7/16/2003 1:05:33am
Modified 7/14/2003 9:24:00am
Accessed 7/16/2003 1:12:45am
User: 502 Group: 502

Screenshot of “prog” info in FTK:

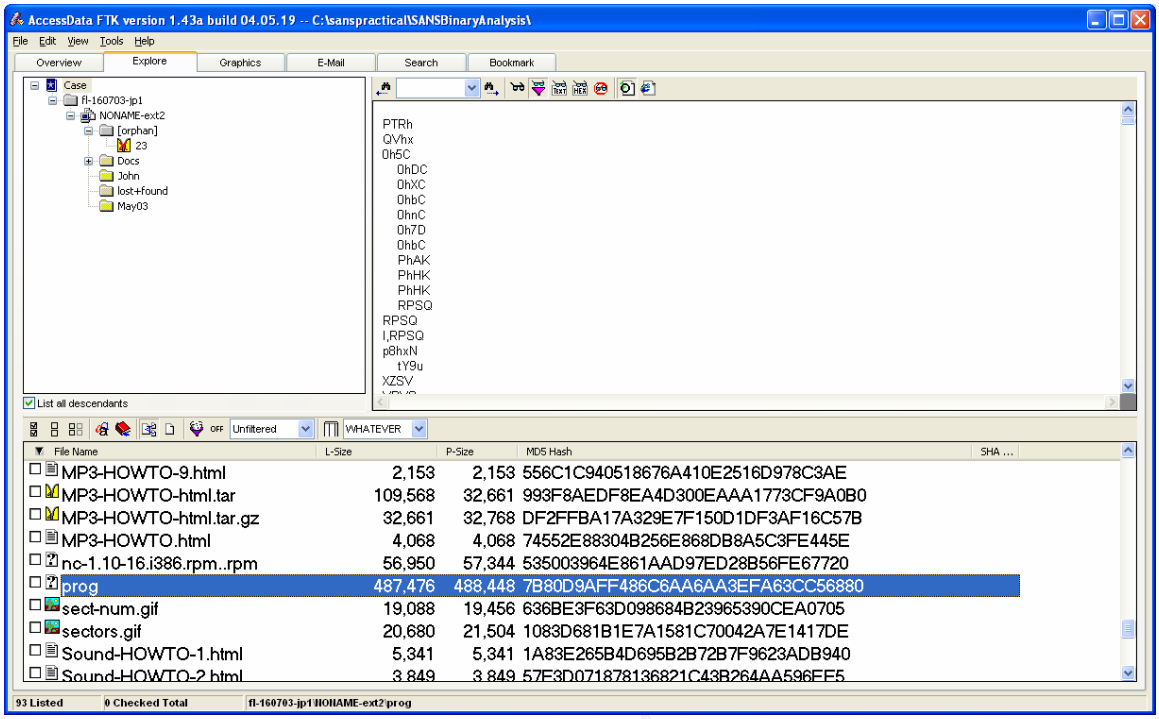


These dates suggest that the file was created (most likely on a different host) on 7/14/2003, then subsequently copied to the floppy on 7/16/2003, then accessed approximately 7 minutes later on 7/16/2003. Of course these times are not completely reliable, since a user can manipulate the system time and/or use the ‘touch’ command to modify them. The modification date prior to the inode changed date implies a transfer from another filesystem to the floppy, barring a user tampering with the MAC times via manipulation of the system clock. Scrolling to the right in the FTK view of “prog” shows file size and MD5 hash info:

File size: 487,476 (logical) / 488,448 (physical)

MD5 hash of “prog”: 7B80D9AFF486C6AA6AA3EFA63CC56880

Screenshot from FTK illustrating MD5 hash and file sizes for “prog”:



In order to fully investigate 'prog', it was exported from FTK and installed on the secondary forensics investigation machine, running Redhat Linux. To export a file from FTK, simply right click on the file in the Explorer view.

On the Redhat Linux box, the command:

```
./prog --help
```

yields the help screen for "prog", shown below:

```
prog:1.0.20 (07/15/03) newt
Usage: prog [OPTION]... [<target-filename>]
use block-list knowledge to perform special operations on
files

--doc VALUE
  where VALUE is one of:
  version  display version and exit
  help  display options and exit
  man  generate man page and exit
  sgml  generate SGML invocation info
--mode VALUE
  where VALUE is one of:
  m  list sector numbers
  c  extract a copy from the raw device
  s  display data
```

```
  p  place data
  w  wipe
  chk  test (returns 0 if exist)
  sb  print number of bytes available
  wipe  wipe the file from the raw device
  frag  display fragmentation information for the file
  checkfrag  test for fragmentation (returns 0 if file is
fragmented)
--outfile <filename> write output to ...
--label   useless bogus option
--name    useless bogus option
--verbose be verbose
--log-thresh <none | fatal | error | info | branch |
progress | entryexit> logging threshold ...
--target <filename> operate on ...
```

"prog" is program that allows performs special operations on files using low-level block list information. Essentially, "prog" is useful for storing and retrieving data from the slack space of a file. In non-technical terms, this means that "prog" allows data to be hidden in existing files. Uses might include hiding passwords or lists of contacts.

"prog" operations include introducing data into the slack space of a file, retrieving the data stored in the slack space by the program, wiping the data stored in the slack space, checking to see if a file is fragmented (that is, if its blocks are not contiguously allocated), and listing sector numbers allocated to a file. The program is self-documenting and can generate its own man page and SGML documentation, which were analyzed to further understand the uses of the program. The application is evidently version 1.0.20, dated 7/15/03. The application was last accessed (and thus potentially executed, although simply viewing the contents of the file or executing the 'touch' command on the file would have the same effect on the access timestamp) on 7/16/2003 at 1:12:45am.

All of the operations supported by the application were tested to verify their functionality, with an eye on verifying that the documented uses of the program coincide with the actual operations performed by the program. The "wipe" operation is presumably for securely deleting a file from disk; this operation failed with multiple "write error" messages on the test machine and could not be evaluated further. The "—label", "—name", and "—target" options appear to be specific to a particular execution environment **(this was gleamed from source code for bmap, once it was determined that "prog" is actually bmap—see Section Program Identification for more details)** and were not usable on the test machine. The three most significant options allow data to be inserted into an arbitrary file's slack space, testing for data stored in the slack space of a file, and for the data stored in the slack space to be retrieved. These operations are illustrated below, where a short text string ("forensics is fun") is inserted into "redrose.JPG", a ~500K JPEG image of a rose, the slack space is checked, and the string is finally retrieved.

Storage of a string in slack space:

```
[root@localhost root]# ./prog -s redrose.JPG
stuffing block 558131
file size was: 503344
slack size: 464
block size: 4096
forensics is fun          ← this string typed in by user
```

Checking slack space:

```
[root@localhost root]# ./prog chk --mode chk redrose.JPG
redrose.JPG has slack
```

Retrieval of a string in slack space:

```
[root@localhost root]# ./prog -s redrose.JPG
getting from block 558131
file size was: 503344
slack size: 464
block size: 4096
forensics is fun
```

Further analysis of the application was performed in Section "Program Identification". In particular, strace (system call trace) was used to verify that the application doesn't tamper with system files or perform other "nasty" operations.

Forensic Details

Output of analysis of "prog" using the Unix "strings" command appears in Appendix A. The strings output contains a number of "/dev/XXXX" entries, which are hard-coded in the bmap library used by 'prog' (this was discovered during investigation of the bmap source code, discussed further in the next section). These device entries are omitted in the listing in Appendix A, because they fill approximately 100 pages. Aside from this omission, the strings output is identical to execution of the following command against the executable (which sorts the strings output and removes duplicate lines):

```
# strings prog | sort | uniq
```

The strings analysis is of limited forensic value except to establish that "prog" and "bmap" are similar. More discussion on this issue appears in the next section, Program Identification. One name and email address are revealed in the strings output and not included in the program's help screen: Keld Simonsen → keld@dkuug.dk. A Google search for "Keld Simonsen" reveals that Keld is working on international standardization for the C programming language [6]. This means that the inclusion of his name in the

binary is likely a result of linking against the standard C library. The following test supports this suspicion:

```
# strings /usr/lib/libc.a | grep –i simonsen
```

on the secondary forensics machine running Linux yields the following:

```
Keld Simonsen
C/o Keld Simonsen, Skt. Jorgens Alle 8, DK-1615 KobenHavn V
```

There is little reason to suspect any further connection between Keld Simonsen and the binary.

System call traces in the following section, Program Identification, do not reveal that any system files (other than the target device, e.g., /dev/hda1) are used by 'prog'. No changes to the /dev/XXXX entries are apparent after execution of "prog". Note that system call traces are included in the next section rather than in this section for continuity, since a close scrutiny of the system call traces of "prog" and "bmap" is used to establish that the programs are functionally equivalent.

Forensics "footprints" are discussed in detail in Sections Program Identification and Case Information. These details are not duplicated here, but briefly, "prog"/bmap will store and retrieve the data stored in an individual file's slack space, so one possible method for determining if "prog"/bmap is in use on a system is to examine individual files using "prog"/bmap, to see if stored data is present. The "-s" option in "prog" can be used for this.

Program Identification

A Google search on "use block-list knowledge to perform" revealed that the program is likely a version of "bmap" [1]. It is discussed briefly on several web pages, including an article on Linux data hiding and recovery [2]. A link to a bmap download site (ftp://ftp.scyld.com/pub/forensic_computing/bmap/) exists on [3] but the link is broken. An anonymous ftp attempt to ftp.scyld.com reveals that the "forensic_computing" directory no longer exists. An older version of the bmap software (v. 1.0.17) is available at [4]. This location was found using Google with keywords "bmap" and "forensics". A subsequent search on Google for "bmap-1.0.20.tar.gz" (using the same syntax for the filename as in the older 1.0.17 version [4]) yielded a site where the source tarball could be downloaded [5].

The tarball was expanded using the following command:

```
tar –xzvf bmap-1.0.20.tar.gz
```

The following files are contained in the tarball:

```
bmap-1.0.20/COPYING
bmap-1.0.20/LICENSE
bmap-1.0.20/Makefile
bmap-1.0.20/README
bmap-1.0.20/bclump.c
bmap-1.0.20/bmap.c
bmap-1.0.20/bmap.sgml.m4
bmap-1.0.20/bmap.spec
bmap-1.0.20/dev_builder.c
bmap-1.0.20/include/bmap.h
bmap-1.0.20/include/slacker.h
bmap-1.0.20/index.html
bmap-1.0.20/libbmap.c
bmap-1.0.20/man/man2/libbmap.2
bmap-1.0.20/mft/COPYING
bmap-1.0.20/mft/Makefile
bmap-1.0.20/mft/README
bmap-1.0.20/mft/helper.c
bmap-1.0.20/mft/include/helper.h
bmap-1.0.20/mft/include/info.h
bmap-1.0.20/mft/include/log.h
bmap-1.0.20/mft/include/mft.h
bmap-1.0.20/mft/include/option.h
bmap-1.0.20/mft/log.c
bmap-1.0.20/mft/option.c
bmap-1.0.20/slacker-modules.c
bmap-1.0.20/slacker.c
```

The author and maintainer of bmap is apparently Daniel Ridge, of Scyld Computing Corporation. This is documented in the README and in the source files for "bmap".

Issuing the file command on "prog" reveals that it is a statically linked and stripped executable. An initial compile of "bmap" (by simply issuing "make") resulted in a dynamically linked version, with a size significantly smaller than "prog". **The Makefile was then modified to create a statically linked "bmap" by changing the CFLAGS in the Makefile to include "-static".** The resulting executable was then stripped with the command "strip bmap". File sizes of "prog" and "bmap" still did not match.

Upon execution of the compiled bmap, it's obvious that the applications are not identical, at least cosmetically. While they both carry the 1.0.20 version number, the command line options of 'prog' are significantly shorter (e.g., "-s" or "—mode s" for "prog" compared to "-slack" or "—mode slack" for bmap). To see this, compare the help screen for "prog" in Section Binary Details and the following help screen for the compiled "bmap":

```
bmap:1.0.20 (05/12/04) newt@scyld.com
Usage: bmap [OPTION]... [<target-filename>]
use block-list knowledge to perform special operations on
files

--doc VALUE
  where VALUE is one of:
  version  display version and exit
  help  display options and exit
  man  generate man page and exit
  sgml  generate SGML invocation info
--mode VALUE
  where VALUE is one of:
  map  list sector numbers
  carve  extract a copy from the raw device
  slack  display data in slack space
  putslack  place data into slack
  wipeslack  wipe slack
  checkslack  test for slack (returns 0 if file has slack)
  slackbytes  print number of slack bytes available
  wipe  wipe the file from the raw device
  frag  display fragmentation information for the file
  checkfrag  test for fragmentation (returns 0 if file is
fragmented)
--outfile <filename> write output to ...
--label   useless bogus option
--name    useless bogus option
--verbose be verbose
--log-thresh <none | fatal | error | info | branch |
progress | entryexit> logging threshold ...
--target <filename> operate on ...
```

One possibility is that a user of "bmap" customized the options to reduce typing effort. Even if the application source codes were identical, the file sizes of the executables might not match, due to use of different versions of gcc and the standard C libraries during compilation. MD5 sums were not compared for "prog" and the compiled version of "bmap" because it is known that the files are not the same.

Strings analysis of "bmap" (see Appendix B for the sorted, unique strings obtained) yields results very similar to that for "prog" (strings analysis for "prog" appears in Appendix A). In fact, the most significant differences relate to the help screen (accessible using the "-h" option for either program), since the text on the screens is different.

A functional test combined with use of strace (which traces the systems calls executed by an application) reveal that 'prog' and 'bmap' are essentially the same, however. All of the useful options were evaluated, producing identical results on a set of test files.

Representative system calls traces for 'prog' and 'bmap', with the command lines used to generate them, follow.

Storage of string "Hello" in slack space of a ~500K file "FILE" using 'prog':

Command line:

```
strace –o /tmp/STRACE_PROG_STORE ./prog –p FILE
```

(-o option for strace specifies an output file for the trace; the command line option –p for prog is equivalent to –mode putslack for bmap)

Contents of /tmp/STRACE_PROG_STORE:

```
execve("./prog", ["./prog", "-p", "FILE"], [/* 36 vars */])
= 0
fcntl64(0, F_GETFD)                     = 0
fcntl64(1, F_GETFD)                     = 0
fcntl64(2, F_GETFD)                     = 0
uname({sys="Linux", node="localhost.localdomain", ...}) = 0
geteuid32()                             = 0
getuid32()                              = 0
getegid32()                             = 0
getgid32()                              = 0
brk(0)                                  = 0x80bedec
brk(0x80bee0c)                          = 0x80bee0c
brk(0x80bf000)                          = 0x80bf000
brk(0x80c0000)                          = 0x80c0000
lstat64("FILE", {st_mode=S_IFREG|0644, st_size=532367,
...}) = 0
open("FILE", O_RDONLY|O_LARGEFILE)      = 3
ioctl(3, FIGETBSZ, 0xbffff534)          = 0
lstat64("FILE", {st_mode=S_IFREG|0644, st_size=532367,
...}) = 0
lstat64("/dev/hda1", {st_mode=S_IFBLK|0660,
st_rdev=makedev(3, 1), ...}) = 0
open("/dev/hda1", O_WRONLY|O_LARGEFILE) = 4
ioctl(3, FIGETBSZ, 0xbffff4a4)          = 0
brk(0x80c2000)                          = 0x80c2000
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
```

```
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
```

```
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
ioctl(3, FIBMAP, 0xbffff534)          = 0
```

```
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
ioctl(3, FIBMAP, 0xbffff534)            = 0
write(2, "stuffing block 590750\n", 22) = 22
write(2, "file size was: 532367\n", 22) = 22
write(2, "slack size: 113\n", 16)       = 16
write(2, "block size: 4096\n", 17)      = 17
_llseek(4, 2419715983, [2419715983], SEEK_SET) = 0
read(0, "Hello\n", 113)                 = 6
write(4, "Hello\n", 6)                  = 6
close(3)                                = 0
close(4)                                = 0
_exit(0)                                = ?
```

The bold write statements above illustrate what was displayed on the screen during execution of the application. The bold read statement is a read from standard input of the

data to be stored in slack space. The underlined write statement is writing the string into the file's slack space (following a seek, 2 lines previous).

Storage of string "Hello" in slack space of a ~500K file "FILE" using 'bmap':

Command line:

```
strace –o /tmp/STRACE_BMAP_STORE ./bmap –-mode putslack FILE
```

Contents of /tmp/STRACE_BMAP_STORE:

```
execve("./bmap", ["./bmap", "--mode", "putslack",
"../FILE"], [/* 36 vars */]) = 0
uname({sys="Linux", node="localhost.localdomain", ...}) = 0
fcntl64(0, F_GETFD)                     = 0
fcntl64(1, F_GETFD)                     = 0
fcntl64(2, F_GETFD)                     = 0
geteuid32()                             = 0
getuid32()                              = 0
getegid32()                             = 0
getgid32()                              = 0
brk(0)                                  = 0x80cdf08
brk(0x80cef08)                          = 0x80cef08
brk(0x80cf000)                          = 0x80cf000
lstat64("../FILE", {st_mode=S_IFREG|0644, st_size=532367,
...}) = 0
open("../FILE", O_RDONLY|O_LARGEFILE)   = 3
ioctl(3, FIGETBSZ, 0xbffff1c4)          = 0
lstat64("../FILE", {st_mode=S_IFREG|0644, st_size=532367,
...}) = 0
lstat64("/dev/hda1", {st_mode=S_IFBLK|0660,
st_rdev=makedev(3, 1), ...}) = 0
open("/dev/hda1", O_WRONLY|O_LARGEFILE) = 4
ioctl(3, FIGETBSZ, 0xbffff134)          = 0
brk(0x80d0000)                          = 0x80d0000
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
```

```
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
```

```
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
```

```
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
ioctl(3, FIBMAP, 0xbffff1c4)            = 0
write(2, "stuffing block 925431\n", 22) = 22
write(2, "file size was: 532367\n", 22) = 22
write(2, "slack size: 113\n", 16)       = 16
write(2, "block size: 4096\n", 17)      = 17
_llseek(4, 3790569359, [3790569359], SEEK_SET) = 0
read(0, "Hello\n", 113)                 = 6
write(4, "Hello\n", 6)                  = 6
close(3)                                = 0
close(4)                                = 0
exit_group(0)                           = ?
```

With one exception, differences in the system call traces above for storage of data in slack space are attributable to different compilation environments—they are functionally equivalent. The difference is that a different target block number is seen in the first bold write statement. This is because a copy of the original file "FILE" was used for the "bmap" execution (e.g., a backup of FILE was made before execution of "prog" and after execution of "prog", this file was renamed to "FILE" before execution of "bmap"). No evidence of forensically significant events such as modification of system files by one version but not the other are evident.

Retrieval of string “Hello” from slack space of the file “FILE” using ‘prog’

Command line:

```
strace –o /tmp/STRACE_PROG_SHOW ./prog –s FILE
```

(-s causes prog to show the contents of slack space for a file and is equivalent to –mode slack for bmap)

Contents of /tmp/STRACE_PROG_SHOW:

```
execve("./prog", ["./prog", "-s", "FILE"], [/* 25 vars */])
= 0
fcntl64(0, F_GETFD)                     = 0
fcntl64(1, F_GETFD)                     = 0
fcntl64(2, F_GETFD)                     = 0
uname({sys="Linux", node="localhost.localdomain", ...}) = 0
geteuid32()                             = 0
getuid32()                              = 0
getegid32()                             = 0
getgid32()                              = 0
brk(0)                                  = 0x80bedec
brk(0x80bee0c)                          = 0x80bee0c
brk(0x80bf000)                          = 0x80bf000
brk(0x80c0000)                          = 0x80c0000
lstat64("FILE", {st_mode=S_IFREG|0644, st_size=532367,
...}) = 0
open("FILE", O_RDONLY|O_LARGEFILE)      = 3
ioctl(3, FIGETBSZ, 0xbffff734)          = 0
lstat64("FILE", {st_mode=S_IFREG|0644, st_size=532367,
...}) = 0
lstat64("/dev/hda1", {st_mode=S_IFBLK|0660,
st_rdev=makedev(3, 1), ...}) = 0
open("/dev/hda1", O_RDONLY|O_LARGEFILE) = 4
ioctl(3, FIGETBSZ, 0xbffff6a4)          = 0
brk(0x80c2000)                          = 0x80c2000
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
```

```
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
ioctl(3, FIBMAP, 0xbffff734)          = 0
```

```
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
```

```
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
write(2, "getting from block 925431\n", 26) = 26
write(2, "file size was: 532367\n", 22) = 22
write(2, "slack size: 113\n", 16)       = 16
write(2, "block size: 4096\n", 17)      = 17
_llseek(4, 3790569359, [3790569359], SEEK_SET) = 0
read(4,
"Hello\n\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..
., 113) = 113
write(1,
"Hello\n\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..
., 113) = 113
close(3)                                = 0
close(4)                                = 0
_exit(0)                                = ?
```

The bold write statements above illustrate what was displayed on the screen during execution of the application. The underlined read statement is a read against /dev/hda1 (following a seek 2 lines previous) that retrieves the hidden data from the file slack.

Retrieval of string “Hello” from slack space of the file “FILE” using ‘bmap’

Command line:

```
strace –o /tmp/STRACE_BMAP_SHOW ./bmap –-mode slack FILE
```

Contents of /tmp/STRACE_BMAP_SHOW:

```
execve("./prog", ["./prog", "-s", "FILE"], [/* 25 vars */])
= 0
fcntl64(0, F_GETFD)                     = 0
fcntl64(1, F_GETFD)                     = 0
fcntl64(2, F_GETFD)                     = 0
uname({sys="Linux", node="localhost.localdomain", ...}) = 0
geteuid32()                             = 0
getuid32()                              = 0
getegid32()                             = 0
getgid32()                              = 0
brk(0)                                  = 0x80bedec
brk(0x80bee0c)                          = 0x80bee0c
brk(0x80bf000)                          = 0x80bf000
brk(0x80c0000)                          = 0x80c0000
lstat64("FILE", {st_mode=S_IFREG|0644, st_size=532367,
...}) = 0
open("FILE", O_RDONLY|O_LARGEFILE)      = 3
ioctl(3, FIGETBSZ, 0xbffff734)          = 0
lstat64("FILE", {st_mode=S_IFREG|0644, st_size=532367,
...}) = 0
lstat64("/dev/hda1", {st_mode=S_IFBLK|0660,
st_rdev=makedev(3, 1), ...}) = 0
open("/dev/hda1", O_RDONLY|O_LARGEFILE) = 4
ioctl(3, FIGETBSZ, 0xbffff6a4)          = 0
brk(0x80c2000)                          = 0x80c2000
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
```

```
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
```

```
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
ioctl(3, FIBMAP, 0xbffff734)            = 0
```

```
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
ioctl(3, FIBMAP, 0xbffff734)           = 0
write(2, "getting from block 925431\n", 26) = 26
write(2, "file size was: 532367\n", 22) = 22
write(2, "slack size: 113\n", 16)      = 16
write(2, "block size: 4096\n", 17)     = 17
_llseek(4, 3790569359, [3790569359], SEEK_SET) = 0
read(4,
"Hello\n\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..
., 113) = 113
write(1,
"Hello\n\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0"..
., 113) = 113
close(3)                               = 0
close(4)                               = 0
_exit(0)                               = ?
```

Differences in the system call traces above for retrieval of data from slack space are attributable to different compilation environments—they are functionally equivalent. In particular, no evidence of forensically significant events such as modification of system files by one version but not the other are evident.

Note that in all of the system call traces, only file system operations are performed. That is, all handles correspond to open files. There is no evidence that the program performs any network operations, because there are no I/O operations performed on handles other

than those corresponding to open files. This rules out TCP connections. Further, there are no system calls that could be attributable to UDP or IP packet transmission.

Another interesting fact is that "prog" can only be run as superuser, since it directly accesses the hard drive (refer to the open of "/dev/hda1", for example). This has some legal implications in case the suspect wasn't supposed to have root access on the machines he had access to.

One critically important point is that use of "prog" (or "bmap") does not change file timestamps. To see this, consider the following three command sets, each of which displays the system date, executes stat to show the MAC times for the file "FILE", executes a "prog" operation against "FILE", and finally, executes stat again to view the MAC times.

"prog" in 'chk' mode:

```
[root@localhost root]# date ; stat FILE ; ./prog --mode chk FILE ; stat
FILE

Tue Jun  1 14:33:06 CDT 2004
  File: `FILE'
  Size: 532367          Blocks: 1048       IO Block: 4096   Regular
File
Device: 301h/769d       Inode: 278417      Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/
root)
Access: 2004-06-01 14:26:46.000000000 -0500
Modify: 2004-05-26 13:54:00.000000000 -0500
Change: 2004-05-26 13:54:00.000000000 -0500

FILE has slack
  File: `FILE'
  Size: 532367          Blocks: 1048       IO Block: 4096   Regular
File
Device: 301h/769d       Inode: 278417      Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/
root)
Access: 2004-06-01 14:26:46.000000000 -0500
Modify: 2004-05-26 13:54:00.000000000 -0500
Change: 2004-05-26 13:54:00.000000000 -0500
```

"prog" in '-p' mode (to insert data into slack):

```
[root@localhost root]# date ; stat FILE ; ./prog -p FILE ; stat FILE

Tue Jun  1 14:34:45 CDT 2004
  File: `FILE'
  Size: 532367          Blocks: 1048       IO Block: 4096   Regular
File
Device: 301h/769d       Inode: 278417      Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/
root)
```

```
Access: 2004-06-01 14:26:46.000000000 -0500
Modify: 2004-05-26 13:54:00.000000000 -0500
Change: 2004-05-26 13:54:00.000000000 -0500

stuffing block 925431
file size was: 532367
slack size: 113
block size: 4096
hi
  File: `FILE'
  Size: 532367          Blocks: 1048       IO Block: 4096   Regular
File
Device: 301h/769d       Inode: 278417      Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/
root)
Access: 2004-06-01 14:26:46.000000000 -0500
Modify: 2004-05-26 13:54:00.000000000 -0500
Change: 2004-05-26 13:54:00.000000000 -0500
```

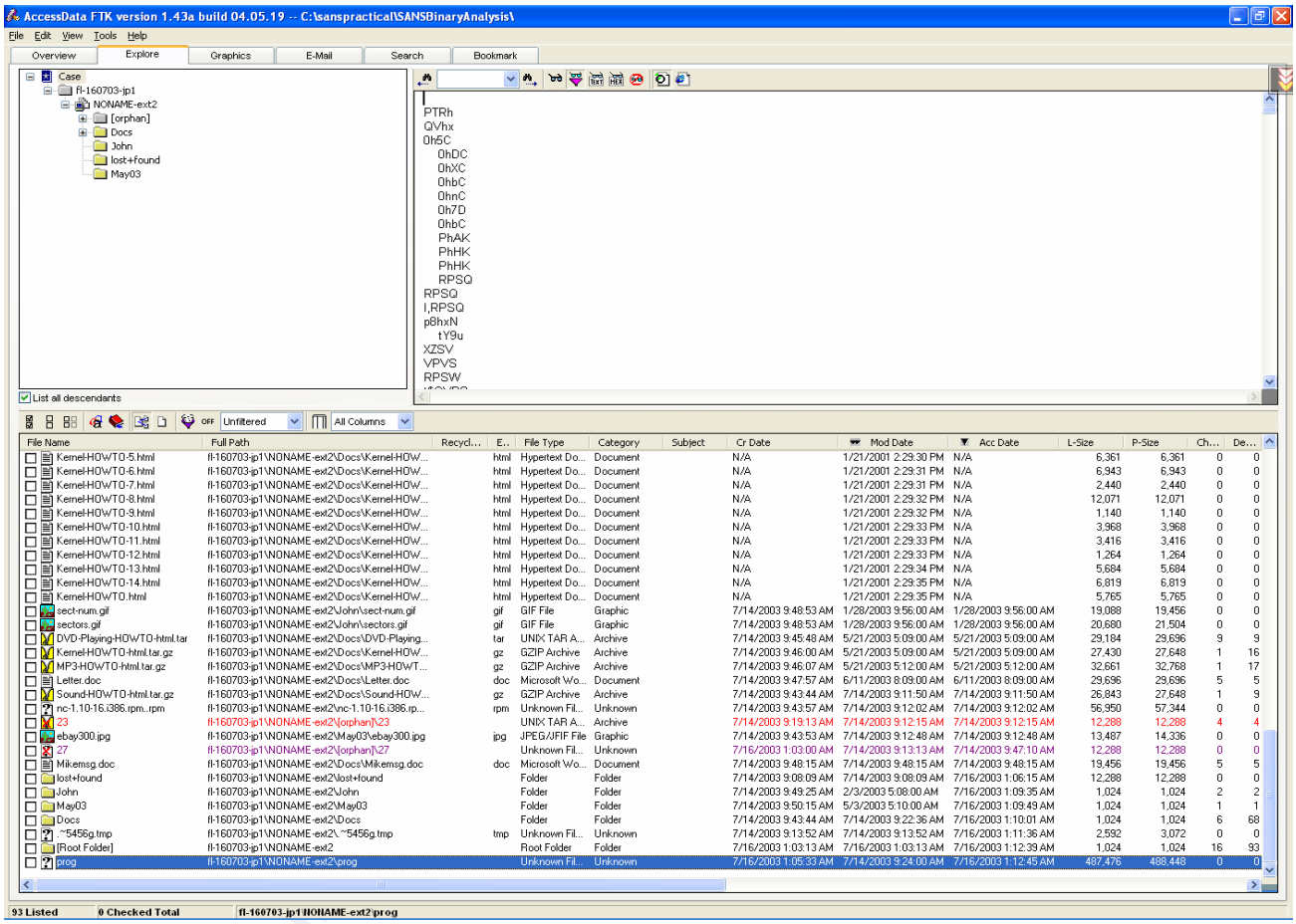
"prog" in '-s' mode (to display contents of slack):

```
[root@localhost root]# date ; stat FILE ; ./prog -s FILE ; stat FILE

Tue Jun  1 14:35:37 CDT 2004
  File: `FILE'
  Size: 532367          Blocks: 1048       IO Block: 4096   Regular
File
Device: 301h/769d       Inode: 278417      Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/
root)
Access: 2004-06-01 14:26:46.000000000 -0500
Modify: 2004-05-26 13:54:00.000000000 -0500
Change: 2004-05-26 13:54:00.000000000 -0500

getting from block 925431
file size was: 532367
slack size: 113
block size: 4096
hi
lo
  File: `FILE'
  Size: 532367          Blocks: 1048       IO Block: 4096   Regular
File
Device: 301h/769d       Inode: 278417      Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/
root)
Access: 2004-06-01 14:26:46.000000000 -0500
Modify: 2004-05-26 13:54:00.000000000 -0500
Change: 2004-05-26 13:54:00.000000000 -0500
```

Thus file access times cannot be used to determine whether "prog" (or "bmap") have been used on a file.

Legal Implications

It is not possible to establish that this version of the program was actually executed given only the evidence on the floppy. The access time of 'prog' is the latest access time on the entire floppy, as illustrated in the FTK screenshot below (where all files present on the floppy, including deleted files, are sorted by access time):



The user could simply have issued the 'touch' command on the executable. However, it seems likely that the suspect executed "prog". Unfortunately, as outlined in the last section, "prog"/"bmap" do not change file access times for target files. In order to determine if data has been stored in slack space on the floppy, the contents of the floppy were evaluated as follows:

An image of the floppy was mounted read-only using the loopback device on the secondary forensics box, running Linux:

```
# mount –r /mnt/usbkey/fl-160703-jp1.dd /mnt/loop –o loop
```

"prog" was then run against each file on the floppy, with the output redirected to a file. For example:

```
# ./prog –p file /mnt/loop/Docs/Mikemsg.doc > FILE
```

The contents of FILE was then evaluated either visually and with the "file" command. One significant hit was found, on the file "Docs/Sound-HOWTO-html.tar.gz", which generated an 805 byte output from "prog". Executing "file" on this output revealed a gzip-compresssed file:

```
# file FILE

FILE: gzip compressed data, was "downloads" from Unix
```

FILE was then renamed to "downloads.gz" and uncompressed using gzip –d:

```
# mv FILE downloads.gz
# gzip –d downloads.gz
```

The file "downloads" contains ASCII text:

```
Ripped MP3s - latest releases:

www.fileshares.org/
www.convenience-city.net/main/pub/index.htm
emmpeethrees.com/hidden/index.htm
ripped.net/down/secret.htm

***NOT FOR DISTRIBUTION***
```

This provides evidence that the suspect might have been using these sites to download ripped MP3s. If he did, then he violated Copyright Act of 1976 as well as the newer Digital Millennium Copyright Act (DMCA). In particular, even if the suspect owned the original sources for the MP3s, he is not entitled to distribute them freely. The presence of a list of MP3 sites also supports the theory that the suspect might have used "prog" (although another individual could have used "bmap" on the floppy, since they are functionally equivalent). The use of prog/bmap to simply hide ASCII text, however, would not specifically violate any internal policies of my organization—we have no restrictions on the use of data hiding software and there is no evidence of copyrighted material on the floppy. Beyond that, simply storing non-copyrighted ASCII text would not violate any other U.S. laws, either. **If additional evidence can be obtained (e.g., network traces) that illustrate that MP3s were downloaded from these sites, then a much stronger case could be made.**

Note: Given a larger number of files to check, it would make sense to develop a short script to automate the use of "prog"/"bmap" and evaluation of the output for each file. This could be written easily in Perl. Given the small number of files, manual evaluation was sufficient.

Other Evidence

A Microsoft Word document with filename "/Docs/Mikemsg.doc" and MD5 hash 82D58D80782A3C017738D00D3A33E2B9 contains the following message:

Hey Mike,

I received the latest batch of files last night and I'm ready to rock-n-roll (ha-ha).

I have some advance orders for the next run. Call me soon.

JP

Analysis of the metadata for the Word document in FTK reveals that the document was created by John Price and its original location was "C:\Documents and Settings\Administrator\Desktop\Mikemsg.doc". This information can be obtained either by viewing the Word file in "Raw Text Format" in FTK or by examining the associated OLE stream containing this info, which FTK provides as a separate item. The pun on "rock-n-roll" suggests that digital media is being transferred or sold.

A number of documentation files exist on the floppy, explaining how to configure playing of MP3s and DVDs under Linux.

An RPM for netcat 1.10-16 is present on the floppy. netcat can be used for quickly establishing client/server connections for file transfers, remote command execution, etc. It could potentially have been used for transfer of files to other machines.

Case Information

There is evidence that indicates company resources were being used by John Price to distribute copyrighted material, but additional evidence would be useful. A program "prog" was installed on the evidence floppy that allows storage of arbitrary data in the slack space of files. In non-technical terms, the program allows arbitrary data to be hidden in existing files. This program, "prog", (or it's functionally equivalent relative, "bmap") was used to hide a list of MP3 download sites on the floppy. The available evidence includes this list of MP3 download sites, a letter to "Mike" implying that a new batch of material is available and that some charge is being made for the files. Further,

the presence of a netcat package, which allows network connections to be established between machines and files to be transferred, is suspicious (unless use of netcat is within the scope of John Price's duties). If network traces (or a backup of the wiped hard drive) were available, a much more damning case might be assembled.

More information on testing for use of "prog"/"bmap"

Alone, the "prog"/"bmap" application has limited usefulness, since the size of the slack space for a single file is limited. It was sufficient in this case for the user to hide a list of MP3 download sites. Another application included in the bmap distribution, called "slacker", allows the aggregate slack space of a directory to be used for storage and retrieval of large amounts of data. Slacker can recursively walk a directory tree, using the available slack space of each file for storage of data. For example, the command:

```
# slacker –mode fill /mnt/floppy < index.html
```

will store the contents of the file "index.html" in the aggregate slack space of all the files on a mounted floppy drive. The command

```
# slacker –mode our /mnt/floppy > retrieved_index.html
```

will retrieve the stored data and write it to a file "retrieved_index.html".

bmap will retrieve the data stored in an individual file's slack space, so one possible method for determining if bmap/slacker are in use on a machine would be to examine individual files using bmap, to see if stored data is present. This is the method used in the present investigation. A much easier method may be available using slacker, however. Consider the following command:

```
[root@localhost bmap-1.0.20]# ./slacker --mode capacity
/mnt/usbkey/recipes
```

This command evaluates the aggregate slack storage capacity of the directory "/mnt/usbkey/recipes", which is a collection of recipes stored on a USB thumbdrive:

```
examining /mnt/usbkey/recipes/almondapricotbiscotti.htm
slack bytes: 4
examining /mnt/usbkey/recipes/coldcarrotsoup.htm
slack bytes: 287
examining /mnt/usbkey/recipes/corbybiscotti.htm
slack bytes: 472
examining /mnt/usbkey/recipes/datebars.htm
slack bytes: 415
examining /mnt/usbkey/recipes/escarolesoup.htm
slack bytes: 124
examining /mnt/usbkey/recipes/mushroomsorzo.htm
slack bytes: 55
```

```
examining /mnt/usbkey/recipes/orzorisotto.htm
slack bytes: 248
examining /mnt/usbkey/recipes/salad.htm
slack bytes: 430
examining /mnt/usbkey/recipes/salad_files
examining /mnt/usbkey/recipes/salad_files/ad1.js
slack bytes: 7
examining /mnt/usbkey/recipes/orzorisotto_files
examining /mnt/usbkey/recipes/orzorisotto_files/ad1.js
slack bytes: 7
examining /mnt/usbkey/recipes/mushroomsorzo_files
examining /mnt/usbkey/recipes/mushroomsorzo_files/ad1.js
slack bytes: 7
examining /mnt/usbkey/recipes/escarolesoup_files
examining /mnt/usbkey/recipes/escarolesoup_files/ad1.js
slack bytes: 7
examining /mnt/usbkey/recipes/datebars_files
examining /mnt/usbkey/recipes/datebars_files/ad1.js
slack bytes: 7
examining /mnt/usbkey/recipes/corbybiscotti_files
examining /mnt/usbkey/recipes/corbybiscotti_files/res0.gif
slack bytes: 448
examining /mnt/usbkey/recipes/corbybiscotti_files/res1.gif
slack bytes: 344
examining /mnt/usbkey/recipes/corbybiscotti_files/res2.gif
slack bytes: 510
examining /mnt/usbkey/recipes/corbybiscotti_files/res3.gif
slack bytes: 308
examining /mnt/usbkey/recipes/coldcarrotsoup_files
examining /mnt/usbkey/recipes/coldcarrotsoup_files/ad1.js
slack bytes: 7
examining /mnt/usbkey/recipes/almondapricotbiscotti_files
examining
mnt/usbkey/recipes/almondapricotbiscotti_files/ad1.js
slack bytes: 7

unformatted capacity: 3694
formatted capacity: 3553
unformatted free: 3641
formatted free: 3553
```

The aggregate storage available is 3553 bytes, all of which are currently available. This seems to indicate that bmap/slacker (or a similar mechanism) have not been used on the system.

The following command was then used to store the contents of a 913 byte HTML file "index.html" in the available slack space:

```
[root@localhost bmap-1.0.20]#./slacker --mode fill
/mnt/usbkey/recipes < index.html
```

A subsequent "capacity" test reveals that some of the capacity has been used (the output is abbreviated to show only the summary information):

```
[root@localhost bmap-1.0.20]# ./slacker --mode capacity
/mnt/usbkey/recipes

<<some output not shown>>
unformatted capacity: 3694
formatted capacity: 3553
unformatted free: 2467
formatted free: 2403
```

This provides an interesting test for a systems administrator to apply to determine if bmap/slacker has been used, since slacker can be applied to entire directory trees. This hypothesis needs further validation, since it is possible that some other utilities might interfere with the use of slacker in this manner. Due to time constraints, this validation was not undertaken.

Interview Questions

1. Who is "Mike"? What is the nature of the batch of files you mention in your message to Mike?

2. Have you ever used data hiding software? Have you ever heard of "bmap"?

3. Have you ever used "ncat"?

4. Why did you wipe the hard drive of your desktop?

5. Why were you interested in MP3 and DVD playing under Linux?

6. Have you ever downloaded files from www.fileshares.org or www.ripped.net?

Additional Information

See references, below. All of these references are cited in previous sections.

References

[1] "LWN.net: Your Linux Info Source Announcements Page", http://lwn.net/2000/0420/announce.php3, alive as of May 26, 2004.

[2] Anton Chuvakin, "Linux Data Hiding and Recovery," http://www.madchat.org/crypto/stegano/unix/wipe/data-hiding-forensics.html, alive as of May 26, 2004.

[3] Electronic Evidence Information Center, http://www.e-evidence.info/other.html, alive as of May 26, 2004.

[4] http://www.securityfocus.com/data/tools/bmap-1.0.17.tar.gz, alive as of May 26, 2004.

[5] http://garchive.movealong.org/bmap-1.0.20/bmap-1.0.20.tar.gz, alive as of May 26, 2004.

[6] WG14, International Standardization Working Group for C, http://std.dkuug.dk/jtc1/sc22/wg14.

Appendix A: 'strings' analysis of 'prog'

```
<[^_]
(^_]
[^_]
[]^_
}.;]
$[^_]
\[^_]
0000000000000000
0123456789abcdefghijklmnopqrstuvwxyz
0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ
 !"#$%&'()*+,-
./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
07/15/03
%[^0-9,+-]
0h5C
0h7D
0h A
0h@A
0h B
0hbC
0h_C
0h&C
0h`D
0hDC
0he=
0h-E
0hm>
0hnC
0hT=
0hXC
0j$P
<0/t
0t+PSj
0t'QSj
```

```
0t'RSj
0t(RVj
0tsQSj
0tsRVj
0<      v
0<      w
1.0.20 (07/15/03)
10646-1:1993// ISO-10646/UCS4/
10646-1:1993/UCS4/ ISO-10646/UCS4/
1997-12-20
2I%%
2PVS
^2XX%
3?Cy
~3SVRR
+45 3122-6543
+45 3325-6543
4$tk
8^_]
8-tx
9$t.
9T00w
9=u>A
%a %b %e %H:%M:%S %Y
%a %b %e %H:%M:%S %Z %Y
Ac+;
AC;]
Accessing a corrupted shared library
AC;M
acpi
Address already in use
Address family not supported by protocol
Advertise error
AELD
;AELD
AFJy
AF<:te1
alias
alnum
alpha
[Am-
amd3d
%a%N%f%N%d%N%b%N%s %h %e %r%N%C-%z %T%N%c%N
ANSI_X3.4-1968
ANSI_X3.4-1968//TRANSLIT
ANSI_X3.4-1986// ANSI_X3.4-1968//
ANSI_X3.4// ANSI_X3.4-1968//
Any of the valid values for \fB--%s\fR can be supplied directly as options. For instance,
\fB--%s\fR can be used in place of \fB--%s=%s\fR.
apic
April
Arena %d:
arg matches against %s
Argument list too long
argv[%d] is NULL
argv[%d] (%s) is not an option
=ascii->INTERNAL
AT_HWCAP:
Attempting to link in too many shared libraries
August
autogenerate document ...
!{>;b
Bad address
Bad file descriptor
Bad font file format
Bad message
be verbose
BG;U
binding file %s to %s: %s symbol `%s'
blank
Block device required
```

```
block size: %d
blue
bmap_get_block_count
bmap_get_block_size
bmap_get_slack_block
bmap_map_block
bmap_raw_close
bmap_raw_open
bogowipe
@bQs
branch
Brazil
Broken pipe
.B %s
B</t
<bt!<b
Bt(P
C09U
+%c %a %l
calling fini: %s
calling init: %s
calling preinit: %s
Can not access a needed shared library
cannot allocate dependency list
Cannot allocate memory
cannot allocate memory for program header
cannot allocate name record
cannot allocate symbol search list
cannot allocate version reference table
Cannot assign requested address
cannot change memory protections
cannot create cache for search path
cannot create RUNPATH/RPATH copy
cannot create scope list
cannot create searchlist
cannot create search path array
cannot create shared object descriptor
cannot dynamically load executable
Cannot exec a shared library directly
cannot extend global scope
cannot load auxiliary `%s' because of empty dynamic string token substitution
cannot make segment writable for relocation
cannot map zero-fill pages
cannot open shared object file
cannot read file data
cannot restore segment prot after reloc
Cannot send after transport endpoint shutdown
cannot stat shared object
C +C
C,+C$)
,ccs=
CDHP
C$+E
Channel number out of range
charset=
checkfrag
checking against %s
checking for version `%s' in file %s required by file %s
clflush
closing file=%s; opencount == %u
cmov
cntrl
C/o Keld Simonsen, Skt. Jorgens Alle 8, DK-1615 Kobenhavn V
Communication error on send
computed block count: %d
Connection refused
Connection reset by peer
Connection timed out
CP367// ANSI_X3.4-1968//
/cpuinfo
C(PV
```

```
CSASCII// ANSI_X3.4-1968//
CSUCS4// ISO-10646/UCS4/
C t
CTPV
;C tU
C<      w+
CX9C
<%d>
[%d]
 %d %d
December
</descrip>
<descrip>
Destination address required
/dev/aztcd
/dev/bpcd
/dev/cdu31a
/dev/cdu535
/dev/cm205cd
/dev/cm206cd
/dev/console
/dev/fd0
[remainder of /dev/XXXX entries deleted]
dI@B
digit
Directory not empty
Disk quota exceeded
display data
display fragmentation information for the file
display options and exit
display version and exit
'_Djz
dlopen
%d %s
DST not allowed in SUID/SGID programs
  dynamic: 0x%0*lx  base: 0x%0*lx   size: 0x%0*Zx
DYNAMIC LINKER BUG!!!
ELF file ABI version invalid
ELF file data encoding not big-endian
ELF file data encoding not little-endian
ELF file OS ABI invalid
ELF file's phentsize not the expected size
ELF file version does not match current one
ELF file version ident does not match current one
ELF load command address/offset not properly aligned
ELF load command alignment not page-aligned
empty dynamics string token substitution
empty dynamic string token substitution
enter
    entry: 0x%0*lx  phdr: 0x%0*lx  phnum:   %*u
entryexit
error
error getting block count
error mapping block %d. block returned 0
error mapping block %d. ioctl failed with %s
error mapping block %d (%s)
error while loading shared libraries
~ERS
/etc/fstab
/etc/ld.so.cache
/etc/localtime
/etc/mtab
/etc/suid-debug
examining a filename or url!
examining an enum!
examining a venum!
Exchange full
Exec format error
exit
extract a copy from the raw device
F0+V
```

```
failed to map segment from shared object
FAJy
fatal
FATAL: cannot determine library version
FATAL: kernel too old
/FBH~
 | \fB%s\fR
       \fB%s\fR
\fB\-\-%s\fR \fIARG\fR %s
\fB\-\-%s\fR \fIFILENAME\fR %s
\fB\-\-%s\fR \fIINT\fR %s
\fB\-\-%s\fR \fIVALUE\fR %s
\fB\-\-%s\fR %s
       \fB%s\fR %s
       \fBSHORTHAND INVOKATION:\fR
FB;u
fd has no blocks
FDHP
@F;E
February
{fG5
File descriptor in bad state
File exists
File name too long
file=%s;  generating link map
file size was: %ld
# File: %s  Location: %Ld  size: %d
file=%s;  needed by %s
file=%s;  needed by %s (relocation dependency)
filesystem reports 0 blocksize
File too large
file too short
Filters not supported with LD_TRACE_PRELINKING
find library=%s; searching
[\fIOPTION\fR]...
\fIVALUE\fR can be one of:
flag-
flagized option invokation
F(Pj
frag
free(): invalid pointer %p!
Friday
F<:t
Function not implemented
fxsr
F\xX
/GBH~
gconv
gconv_end
gconv_init
gconv-modules
GCONV_PATH
gconv_trans
gconv_trans_context
gconv_trans_end
gconv_trans_init
@G;E
generate man page and exit
generate SGML invocation info
getting from block %d
gfff
G +G
glibc-ld.so.cache1.1
gmon
graph
green
&gt;</tag> %s
<GtZ<gt.
Gu~1
Gu[1
Gu#1
```

```
GuL1
GuP1
h`?
h=>
h-?
h!>
h[?
h@=
h$?
H^_]
h7?
hbK
hCJ
h,D
hdI
help
heO
%h %e %T
hg?
hg@
hH?
hhC
hHK
h@I
hI?
h`J
h)J
h K
hK=
hKM
h L
hLK
h`M
h M
h:M
%H:%M
%H:%M:%S
hn?
h_N
h@N
h@O
HOSTALIASES
Host is down
how did we get here?
hP>
hs?
H%T=
hU?
%hu:%hu:%hu
huK
hxI
hy?
hzC
i18n:1999
i386
i486
i586
i686
I9C-
ia64
IBM367// ANSI_X3.4-1968//
Identifier removed
I!G.
IGNORE
Illegal seek
/.../image
%I:%M:%S %p
Inappropriate ioctl for device
info
inity
Input/output error
```

```
=INTERNAL->ascii
internal error
=INTERNAL->ucs2
=INTERNAL->ucs2reverse
=INTERNAL->ucs4
=INTERNAL->ucs4le
=INTERNAL->utf8
Interrupted system call
Interrupted system call should be restarted
in use bytes     = %10u
Invalid argument
Invalid cross-device link
invalid ELF header
Invalid exchange
invalid index %d
invalid mode for dlopen()
invalid option: %s
Invalid or incomplete multibyte or wide character
Invalid request code
Invalid request descriptor
Invalid slot
invalid value for enum
I,RPSQ
Is a directory
Is a named type file
ISO-10646// ISO-10646/UCS4/
ISO-10646/UTF-8/ ISO-10646/UTF8/
ISO_646.IRV:1991// ANSI_X3.4-1968//
ISO646-US// ANSI_X3.4-1968//
ISO/IEC 14652 i18n FDCC-set
ISO/IEC JTC1/SC22/WG20 - internationalization
ISO-IR-193// ISO-10646/UTF8/
ISO-IR-6// ANSI_X3.4-1968//
~j2=
January
/j     hs
@j/P
Jt}1
Jtd1
Jte1
Jtl1
Jt.P
JtPG
JtTG
July
June
keld@dkuug.dk
Keld Simonsen
kpnJ
L[^_]
label
LANG
LANGUAGE
 (lazy)
LC_ADDRESS
LC_ALL
LC_COLLATE
LC_CTYPE
LC_IDENTIFICATION
LC_MEASUREMENT
LC_MESSAGES
LC_MONETARY
LC_NAME
LC_NUMERIC
LC_PAPER
LC_TELEPHONE
LC_TIME
LC_XXX
LD_AOUT_LIBRARY_PATH
LD_AOUT_PRELOAD
LD_BIND_NOT
```

```
LD_BIND_NOW
LD_DEBUG_OUTPUT
LD_DYNAMIC_WEAK
LD_LIBRARY_PATH
LD_ORIGIN_PATH
LD_PRELOAD
LD_PROFILE
L       dQ
ld.so-1.7.0
LD_WARN
Level 2 halted
Level 2 not synchronized
Level 3 halted
Level 3 reset
/lib/
libc
.lib section in a.out corrupted
Link has been severed
Link number out of range
list sector numbers
load auxiliary object=%s requested by file=%s
load filtered object=%s requested by file=%s
LOCALDOMAIN
/locale.alias
LOCPATH
logging threshold ...
log-thresh
lower
 [&lt;%s-filename&gt;]
}+;M
|M9u
Machine is not on the network
main
<main program>
MALLOC_CHECK_
malloc: top chunk is corrupt
MALLOC_TRACE
malloc: using debugging hooks
mapping block %lu
March
matched against an enum val
matched against an venum val
matches against %s
max mmap bytes   = %10lu
max mmap regions = %10u
%m/%d/%y
MemFree: %ld kB
/meminfo
MemTotal: %ld kB
messages
Message too long
mft_getopt
mft_log_init
mft_log_shutdown
MFT_LOG_THRESH
M%hu.%hu.%hu%n
MMAP_MAX_
MMAP_THRESHOLD_
mode
module
Monday
MP0!
mtrr
Multihop attempted
[!|n
name
Name not unique on network
nbd-server
Network dropped connection on reset
Network is down
Network is unreachable
```

```
newt
(nil)
NLSPATH
^[nN]
No anode
No buffer space available
No child processes
No CSI structure available
No data available
no filename. try '--help' for help.
no index
No locks available
No medium found
No message of desired type
none
normal
No route to host
No space left on device
No such device
No such device or address
No such file or directory
No such process
Not a directory
Not a XENIX named type file
 not defined in file
' not found (required by
November
no version information available (required by
 (no version symbols)
No XENIX semaphores available
nplurals=
nul block while mapping block %d.
(null)
NULL filename supplied
NULL value for slack_block
Numerical argument out of domain
Numerical result out of range
|;#o
object file has no dynamic section
Object is remote
October
off_t too small!
 of Verdef record
 of Verneed record
only ET_DYN and ET_EXEC can be loaded
opening file=%s; opencount == %u
operate on ...
Operation already in progress
Operation not permitted
Operation not supported
Operation now in progress
operation to perform on files
orange
ORIGIN
OSF00010020// ANSI_X3.4-1968//
OSF00010100// ISO-10646/UCS2/
OSF00010101// ISO-10646/UCS2/
OSF00010102// ISO-10646/UCS2/
OSF00010104// ISO-10646/UCS4/
OSF00010105// ISO-10646/UCS4/
OSF00010106// ISO-10646/UCS4/
OSF05010001// ISO-10646/UTF8/
outfile
out of memory
out of memory [
Out of memory while initializing profiler
Out of streams resources
OUTPUT_CHARSET
P0+H
p8hxN
Package not installed
```

```
parse error
parser stack overflow
Permission denied
Ph @
Ph"@
Ph4n
PhAK
Ph|b
PhHK
PhI?
place data
PLATFORM
plural=
$po?b
POSIX
posixrules
print
print number of bytes available
proc
/proc
process_match
processor
/proc/self/cwd
/proc/self/exe
/proc/sys/kernel/osrelease
.profile
prog
<program name unknown>
progress
protected
Protocol driver not attached
Protocol error
Protocol family not supported
Protocol not available
Protocol not supported
Protocol wrong type for socket
~pRSV
~>PS
pse36
PSh|
PSj0
@PSR
PSRW
~"PSV
PSVj
%p%t%g%t%m%t%f
PTRh
P ;U
punct
PVh|
PVRS
PWVS
~!Q+B
Qh`b
Qj      h_
Qj      hh
QPRW
QPSV
QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQj
QSh|
QSj
QSj0
QSVW
QVh|
QVhx
QVj
QVj0
QVWP
QWVS
)r+[
raw fd is %d
```

```
read error
Read-only file system
realloc(): invalid pointer %p!
relocation error
relocation processing: %s%s
Remote address changed
Remote I/O error
Report bugs to %s.
RESOLV_HOST_CONF
RES_OPTIONS
Resource deadlock avoided
Resource temporarily unavailable
RFS specific error
Rh`b
R Iu
Rj@WS
RPATH
RPh`
RPSQ
RPSW
RPWV
RSj
RSj0
RSVP
RSWV
RUNPATH
RVj
RWVS
> %s
 | %s
 [%s]
--%s <
               (%s)
s=;]
--%s <arg> %s
Saturday
%s: cannot create file: %s
%s: cannot map file: %s
%s: cannot open file: %s
%s: cannot stat file: %s
%s does not have fragmentation
%s does not have slack
 search cache=%s
 search path=
seconds
seek error
seek failure
September
%s: file is no correct profile data file for `%s'
 [<%s-filename>]
--%s <filename> %s
%s fragmented between %d and %d
               (%s from file %s)
sgml
shared object cannot be dlopen()ed
shared object not open
%s has fragmentation
%s has holes in excess of %ld bytes...
%s has multiple links.
%s has slack
.SH DESCRIPTION
.SH NAME
.SH REPORTING BUGS
.SH SYNOPSIS
--%s <int> %s
%s is a well-formed argument
%s is not a regular file.
Sj:P
slack size: %d
Socket operation on non-socket
Socket type not supported
```

```
Software caused connection abort
=S@P
space
%s: profiler found no PLTREL in object %s
%s: profiler out of memory shadowing PLTREL of %s
SPWQ
SQRP
Srmount error
SRVW
  %s  %s
--%s %s
--%s   %s
%s \- %s
%s: %s
sse2
%s:%s %s
%s %s %s %s %d %d
%s: %s: %s%s%s%s%s
%s: Symbol `%s' has different size in shared object, consider re-linking
;S t<
Stale NFS file handle
stat reports %d blocks: %d
st      C
Streams pipe error
Structure needs cleaning
stuffing block %d
s=;u
Success
Sunday
--%s VALUE
SVj0
SWh|
SWVV
symbol
symbol=%s;  lookup in file=%s
/SYS
/SYS_
syslog: unknown facility/priority: %x
system bytes     = %10u
system search path
</t$
<*t><*
 @t-
 @t:
;$t,
*<'t
t-;]
t';]
t       ;}
<$t0
t0@Nt
t0QV
@t0R
t0@t
t0tv
 t0Wj
t4Qj
 @t5
@t5=
t";5
;:t7G
t8<:u4
t       9P
<table bgcolor=%s><tr><td>%s: %s</td></tr></table><br>
<table bgcolor=%s><tr><td>%s</td></tr></table><br>
<table bgcolor=%s><tr><td></td></tr></table><br>
<tag>--%s &lt;
<tag>--%s &lt;arg&gt;</tag> %s
<tag>--%s &lt;filename&gt;</tag> %s
<tag>--%s &lt;int&gt;</tag> %s
<tag>--%s</tag> %s
```

```
<tag>--%s</tag>       %s
<tag>%s</tag>  %s
<tag>--%s VALUE</tag>
tANt:
target
target file block size: %d
target filename: %s
t       B<:u
tB;u
tcB9
tC;E
tCVS
t);E
test for fragmentation (returns 0 if file is fragmented)
test (returns 0 if exist)
Text file busy
tHRV
.TH %s "%d" "%s" "%s" "%s"
Thursday
Timer expired
tiPh
=t%j
tj/P
tJPVj
tkWQ
t=Ky
tLPj
tLRh`b
TMPDIR
tmPh
<*tm<'ti<Ite
t       N;
tnF;5
t(@Nt
t(Nu
;:toG
tolower
Too many levels of symbolic links
Too many links
Too many open files
Too many open files in system
Too many references: cannot splice
Too many users
TOP_PAD_
Total (incl. mmap):
toupper
t%Pj
t PS
t$PS
t&PS
t PVj
tQ9u
t`Qh`b
t>Qj
t/Qj
_tQOt
tQRS
t*QVj
t&QVj
t$QVPS
Transport endpoint is already connected
Transport endpoint is not connected
TRIM_THRESHOLD_
true
t,RVWP
try '--help' for help.
  trying file=%s
tS;}
t SVj
</tt>
?/tt
```

```
<\t~<\tn
<tt>%s [&lt;OPTIONS&gt;]
<tt>%s</tt> invocation
t(;u
t(;U
Tuesday
t'WS
tY9u
TZDIR
< tZ<   tB<\t2
|,;u
u4@P
u8@P
=u8Q)+
u^9u
UCS-2BE// UNICODEBIG//
=ucs2->INTERNAL
UCS-2// ISO-10646/UCS2/
UCS2// ISO-10646/UCS2/
UCS-2LE// ISO-10646/UCS2/
=ucs2reverse->INTERNAL
UCS-4BE// ISO-10646/UCS4/
=ucs4->INTERNAL
UCS-4// ISO-10646/UCS4/
UCS-4LE//
=ucs4le->INTERNAL
uc;u
uD;s
ug;]
U^h6LU3
Unable to allocate buffer
Unable to determine blocksize
Unable to determine count
unable to determine filesystem blocksize
unable to determine raw device of %s
Unable to open file: %s
unable to open raw device %s
unable to raw open %s
unable to stat fd
Unable to stat fd
Unable to stat file: %s
unable to stat raw device %s
undefined symbol:
unexpected PLT reloc type 0x
unexpected reloc type 0x
UNICODELITTLE// ISO-10646/UCS2/
Universal
Unknown error
unspecified
unsupported version
u @P
u,@P
u$@P
upper
Usage: %s [OPTION]...
US-ASCII// ANSI_X3.4-1968//
use block-list knowledge to perform special operations on files
useless bogus option
/usr/lib/
/usr/lib/gconv
/usr/lib/gconv/gconv-modules.cache
/usr/lib/locale
/usr/share/locale
/usr/share/zoneinfo
=utf8->INTERNAL
UTF-8// ISO-10646/UTF8/
UTF8// ISO-10646/UTF8/
U.y`
uYD?e
<       v$
<       v;1
```

```
<       v61
v A)
Value too large for defined data type
verbose
version
, version
Vh`b
violet
V$PQ)
VPRQ
VPVS
VQRP
VQSP
~VQSV
VQSW
VRSP
VSh|
VUUU
$v?V
V +V
v)WRj
vwWSQ
<       w[
w!;]
w&;=
w%;]
<       w1
<       w1j
wA;U
WCHAR_T// INTERNAL
weak version `
Wednesday
Wh4n
Where <bf>OPTIONS</bf> may include any of:
  where VALUE is one of:
white
wipe
wipe the file from the raw device
 with link time reference
Wj      hr
Wj@j
w_;M
WQj0
WQRV
write error
write output to ...
Wrong medium type
WRSP
w};u
w%;u
WVj
WVj0
WVS1
WVSQ
WVUS
X^_]
X[^_
xdigit
_Xhs
x~QS
x Rj
*~xx
XZh@
XZSh
XZSV
yellow
%Y-%m-%d
^[yY]
Z[^_]
ZYhs
```

Appendix B: 'strings' analysis of compiled 'bmap'

```
^ +^
~*;]
<[^_
 [^_
_^[]
,[^_
. ].
. {.
[]^_
{.
$[^_
0000000000000000
0123456789abcdefghijklmnopqrstuvwxyz
0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ
 !"#$%&'()*+,-
./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
05/12/04
%[^0-9,+-]
0h ~
0h]~
0h#}
0hB{
0hD}
0hk~
0hp}
0hyz
0h z
<0=t+
0t0QSj
0t)1
0t1PVj
0tB<\
0tbPSj
0tk9
0t$QSj
0t=<\u
0<      v
0<      w
1.0.20 (05/12/04)
10646-1:1993// ISO-10646/UCS4/
10646-1:1993/UCS4/ ISO-10646/UCS4/
1997-12-20
2I%%
<2t"
^2XX%
3?Cy
+45 3122-6543
+45 3325-6543
4?RV
8j$Q
8-ts
8/u^
;9wz
%a %b %e %H:%M:%S %Y
%a %b %e %H:%M:%S %Z %Y
Ac+;
Accessing a corrupted shared library
AC;M
acpi
A %d %d
Address already in use
Address family not supported by protocol
Advertise error
AELD
?AELDt
```

```
AFJy
AG<:tq1
alias
alnum
alpha
[Am-
%a%N%f%N%d%N%b%N%s %h %e %r%N%C-%z %T%N%c%N
ANSI_X3.4-1968
ANSI_X3.4-1968//TRANSLIT
ANSI_X3.4-1986// ANSI_X3.4-1968//
ANSI_X3.4// ANSI_X3.4-1968//
Any of the valid values for \fB--%s\fR can be supplied directly as options. For instance,
\fB--%s\fR can be used in place of \fB--%s=%s\fR.
apic
April
Arena %d:
arg matches against %s
Argument list too long
argv[%d] is NULL
argv[%d] (%s) is not an option
=ascii->INTERNAL
Attempting to link in too many shared libraries
August
autogenerate document ...
a. v.
!{>;b
Bad address
Bad file descriptor
Bad font file format
Bad message
B +B
be verbose
binding file %s to %s: %s symbol `%s'
blank
Block device required
block size: %d
blue
bmap
bmap_get_block_count
bmap_get_block_size
bmap_get_slack_block
bmap_map_block
bmap_raw_close
bmap_raw_open
bogowipe
@bQs
branch
Brazil
Broken pipe
.B %s
B s&
B s#
B<,t
B<:t
B<:tP
Bt(P
C(Ae
CAJy
+%c %a %l
calling fini: %s
calling init: %s
calling preinit: %s
Can not access a needed shared library
cannot allocate dependency list
Cannot allocate memory
cannot allocate memory for program header
cannot allocate name record
cannot allocate symbol search list
cannot allocate version reference table
Cannot assign requested address
cannot change memory protections
```

```
cannot create cache for search path
cannot create RUNPATH/RPATH copy
cannot create scope list
cannot create searchlist
cannot create search path array
cannot create shared object descriptor
cannot dynamically load executable
Cannot exec a shared library directly
cannot extend global scope
cannot handle TLS data
cannot load auxiliary `%s' because of empty dynamic string token substitution
cannot make segment writable for relocation
cannot map zero-fill pages
cannot open shared object file
cannot read file data
cannot restore segment prot after reloc
Cannot send after transport endpoint shutdown
cannot stat shared object
carve
,ccs=
C$+E
Channel number out of range
charset=
checkfrag
checking against %s
checking for version `%s' in file %s required by file %s
checkslack
clflush
closing file=%s; opencount == %u
cmov
cntrl
C/o Keld Simonsen, Skt. Jorgens Alle 8, DK-1615 Kobenhavn V
Communication error on send
computed block count: %d
Connection refused
Connection reset by peer
Connection timed out
continued
CP367// ANSI_X3.4-1968//
C$PS
/cpuinfo
CRSV
CSASCII// ANSI_X3.4-1968//
CSUCS4// ISO-10646/UCS4/
$C;Z r
<%d>
[%d]
D      dP
December
</descrip>
<descrip>
Destination address required
/dev/aztcd
/dev/bpcd
/dev/cdu31a
/dev/cdu535
/dev/cm205cd
/dev/cm206cd
/dev/console
/dev/fd0
[remainder of /dev/XXXX entries deleted]
device mismatch 0x%x != 0x%x
Device not a stream
Device or resource busy
dI@B
digit
Directory not empty
Disk quota exceeded
display data in slack space
display fragmentation information for the file
display options and exit
```

```
display version and exit
Dj/P
'_Djz
D. K.
dlopen
%d %s
DST not allowed in SUID/SGID programs
  dynamic: 0x%0*lx  base: 0x%0*lx   size: 0x%0*Zx
DYNAMIC LINKER BUG!!!
E$!]
ELF file ABI version invalid
ELF file data encoding not little-endian
ELF file OS ABI invalid
ELF file's phentsize not the expected size
ELF file version does not match current one
ELF file version ident does not match current one
ELF load command address/offset not properly aligned
ELF load command alignment not page-aligned
ELFuc
empty dynamics string token substitution
empty dynamic string token substitution
enter
    entry: 0x%0*lx  phdr: 0x%0*lx  phnum:   %*u
entryexit
error
error getting block count
error mapping block %d. block returned 0
error mapping block %d. ioctl failed with %s
error mapping block %d (%s)
error while loading shared libraries
/etc/fstab
/etc/ld.so.cache
/etc/localtime
/etc/mtab
/etc/suid-debug
E$!u
E$!U
examining a filename or url!
examining an enum!
examining a venum!
Exchange full
Exec format error
exit
extract a copy from the raw device
failed to map segment from shared object
FAJy
fatal
FATAL: cannot determine library version
FATAL: kernel too old
 | \fB%s\fR
       \fB%s\fR
\fB\-\-%s\fR \fIARG\fR %s
\fB\-\-%s\fR \fIFILENAME\fR %s
\fB\-\-%s\fR \fIINT\fR %s
\fB\-\-%s\fR \fIVALUE\fR %s
\fB\-\-%s\fR %s
       \fB%s\fR %s
       \fBSHORTHAND INVOKATION:\fR
fd has no blocks
February
F +F
{fG5
File descriptor in bad state
File exists
File name too long
file=%s;  generating link map
file size was: %ld
# File: %s  Location: %Ld  size: %d
file=%s;  needed by %s
file=%s;  needed by %s (relocation dependency)
filesystem reports 0 blocksize
```

```
File too large
file too short
Filters not supported with LD_TRACE_PRELINKING
find library=%s; searching
[\fIOPTION\fR]...
\fIVALUE\fR can be one of:
Fj@V
flag-
flagized option invokation
frag
free(): invalid pointer %p!
Friday
F. T.
Function not implemented
FX9F
fxsr
F\xX
gconv
gconv_end
gconv_init
gconv-modules
GCONV_PATH
gconv_trans
gconv_trans_context
gconv_trans_end
gconv_trans_init
G;=d
generate man page and exit
generate SGML invocation info
getting from block %d
gfff
G +G
G,+G$
glibc-ld.so.cache1.1
gmon
G;=P
graph
green
&gt;</tag> %s
G$+U
Gu~1
Gu[1
Gu#1
GuL1
GuP1
h^}
h=|
h |
h;{
h?{
h'|
h${
h+|
h1|
h2{
h4{
h6}
h9|
h9{
hA~
help
%h %e %T
H +H
H$;H(
hK}
%H:%M
%H:%M:%S
HOSTALIASES
Host is down
how did we get here?
HPWW
```

```
hU{
Hu7R
%hu:%hu:%hu
hw~
h?z
h\z
`. I.
i18n:1999
i386
i486
i586
i686
I9C-
ia64
IBM367// ANSI_X3.4-1968//
Identifier removed
I!G.
IGNORE
Illegal seek
/.../image
%I:%M:%S %p
Inappropriate ioctl for device
info
inity
Input/output error
=INTERNAL->ascii
internal error
=INTERNAL->ucs2
=INTERNAL->ucs2reverse
=INTERNAL->ucs4
=INTERNAL->ucs4le
=INTERNAL->utf8
Interrupted system call
Interrupted system call should be restarted
in use bytes     = %10u
Invalid argument
Invalid cross-device link
invalid ELF header
Invalid exchange
invalid index %d
invalid mode for dlopen()
invalid option: %s
Invalid or incomplete multibyte or wide character
Invalid request code
Invalid request descriptor
Invalid slot
invalid value for enum
I,RPSQ
Is a directory
Is a named type file
ISO-10646// ISO-10646/UCS4/
ISO-10646/UTF-8/ ISO-10646/UTF8/
ISO_646.IRV:1991// ANSI_X3.4-1968//
ISO646-US// ANSI_X3.4-1968//
ISO/IEC 14652 i18n FDCC-set
ISO/IEC JTC1/SC22/WG20 - internationalization
ISO-IR-193// ISO-10646/UTF8/
ISO-IR-6// ANSI_X3.4-1968//
~j2=
J(9M
January
/j     h
j:h
@j/P
J,QS
j. S.
Jti1
Jt.P
Jtv1
July
June
```

```
[. K.
keld@dkuug.dk
Keld Simonsen
kpnJ
label
LANG
LANGUAGE
 (lazy)
LC_ADDRESS
LC_ALL
LC_COLLATE
LC_CTYPE
LC_IDENTIFICATION
LC_MEASUREMENT
LC_MESSAGES
LC_MONETARY
LC_NAME
LC_NUMERIC
LC_PAPER
LC_TELEPHONE
LC_TIME
LD_AOUT_LIBRARY_PATH
LD_AOUT_PRELOAD
LD_ASSUME_KERNEL
LD_BIND_NOT
LD_BIND_NOW
LD_DEBUG_OUTPUT
LD_DYNAMIC_WEAK
LD_LIBRARY_PATH
LD_ORIGIN_PATH
LD_PRELOAD
LD_PROFILE
ld.so-1.7.0
LD_WARN
Level 2 halted
Level 2 not synchronized
Level 3 halted
Level 3 reset
/lib/
libc
.lib section in a.out corrupted
Link has been severed
Link number out of range
list sector numbers
load auxiliary object=%s requested by file=%s
load filtered object=%s requested by file=%s
LOCALDOMAIN
/locale.alias
LOCPATH
logging threshold ...
log-thresh
lower
 [&lt;%s-filename&gt;]
Machine is not on the network
main
<main program>
MALLOC_CHECK_
malloc: top chunk is corrupt
MALLOC_TRACE
malloc: using debugging hooks
mapping block %lu
March
matched against an enum val
matched against an venum val
matches against %s
max mmap bytes   = %10lu
max mmap regions = %10u
%m/%d/%y
MemFree: %ld kB
/meminfo
MemTotal: %ld kB
```

```
messages
Message too long
mft_getopt
mft_log_init
mft_log_shutdown
MFT_LOG_THRESH
M%hu.%hu.%hu%n
MMAP_MAX_
MMAP_THRESHOLD_
mode
module
Monday
MP0!
mtrr
Multihop attempted
[!|n
name
Name not unique on network
nbd-server
Network dropped connection on reset
Network is down
Network is unreachable
newt@scyld.com
(nil)
NLSPATH
^[nN]
No anode
No buffer space available
No child processes
No CSI structure available
No data available
no filename. try '--help' for help.
no index
No locks available
No medium found
No message of desired type
none
normal
No route to host
No space left on device
No such device
No such device or address
No such file or directory
No such process
Not a directory
Not a XENIX named type file
 not defined in file
' not found (required by
November
no version information available (required by
 (no version symbols)
No XENIX semaphores available
nplurals=
N(Qj
nul block while mapping block %d.
(null)
NULL filename supplied
NULL value for slack_block
Numerical argument out of domain
Numerical result out of range
|;#o
. O.
object file has no dynamic section
Object is remote
October
 of Verdef record
 of Verneed record
only ET_DYN and ET_EXEC can be loaded
opening file=%s; opencount == %u
operate on ...
Operation already in progress
```

```
Operation canceled
Operation not permitted
Operation not supported
Operation now in progress
operation to perform on files
orange
ORIGIN
OSF00010020// ANSI_X3.4-1968//
OSF00010100// ISO-10646/UCS2/
OSF00010101// ISO-10646/UCS2/
OSF00010102// ISO-10646/UCS2/
OSF00010104// ISO-10646/UCS4/
OSF00010105// ISO-10646/UCS4/
OSF00010106// ISO-10646/UCS4/
OSF05010001// ISO-10646/UTF8/
outfile
out of memory
out of memory [
Out of memory while initializing profiler
Out of streams resources
OUTPUT_CHARSET
P4uB
p8hx
p8v;
Package not installed
parse error
parser stack overflow
Permission denied
@Ph
Ph@|
Ph|e
Phee
Phpe
Ph s
Phzs
^_Pj
Pj     h
place data into slack
PLATFORM
plural=
pnv!
$po?b
POSIX
posixrules
PQRW
~$PR+}
{$PR)
print
print number of slack bytes available
proc
/proc
process_match
processor
/proc/self/exe
/proc/sys/kernel/osrelease
.profile
<program name unknown>
progress
protected
Protocol driver not attached
Protocol error
Protocol family not supported
Protocol not available
Protocol not supported
Protocol wrong type for socket
PRQSV
PRQV
PRSV
pse36
PSh
PSh#
```

```
PSj
PSj0
@PSR
PSRV
PSVR
PSVW
PSWV
%p%t%g%t%m%t%f
PTRhd
;pTta
P$;U
punct
putslack
p[v.
PVhee
PVj0
PVSW
PWVh
PWVS
Qhpe
Qj      h
Qj@j
QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQj
QShL
QSj
QSj0
QSPV
~)QV
QVh0
QWRV
QWSV
QWVS
. r.
)r+[
raw fd is %d
r{;E
read error
Read-only file system
realloc(): invalid pointer %p!
relocation error
relocation processing: %s%s
Remote address changed
Remote I/O error
Report bugs to %s.
RESOLV_HOST_CONF
RES_OPTIONS
Resource deadlock avoided
Resource temporarily unavailable
RFS specific error
R Iu
RPATH
RPh@
RPSQ
RPWS
RQSV
RQSW
RSj
RSVP
RSWV
RtmVj
RUNPATH
RV)}
RVj0
RVSP
RWVS
> %s
 | %s
 [%s]
--%s <
. s.
               (%s)
```

```
. S.
s1It
--%s <arg> %s
Saturday
%s: cannot create file: %s
%s: cannot map file: %s
%s: cannot open file: %s
%s: cannot stat file: %s
%s does not have fragmentation
%s does not have slack
 search cache=%s
 search path=
seco
seek error
seek failure
September
%s: error: %s: %s (%s)
%s: file is no correct profile data file for `%s'
 [<%s-filename>]
--%s <filename> %s
%s fragmented between %d and %d
               (%s from file %s)
sgml
shared object cannot be dlopen()ed
shared object not open
%s has fragmentation
%s has holes in excess of %ld bytes...
%s has multiple links.
%s has slack
.SH DESCRIPTION
.SH NAME
.SH REPORTING BUGS
.SH SYNOPSIS
--%s <int> %s
%s is a well-formed argument
%s is not a regular file.
Sj     h
slack
slackbytes
slack size: %d
Socket operation on non-socket
Socket type not supported
Software caused connection abort
space
%s: profiler found no PLTREL in object %s
%s: profiler out of memory shadowing PLTREL of %s
SPVQ
SQhU
SQRW
~.SR
~@SR
Srmount error
S(RV
  %s  %s
--%s %s
--%s   %s
%s \- %s
%s: %s
sse2
%s:%s %s
%s %s %s %s %d %d
%s: %s: %s%s%s%s%s
SStJ
SStV
%s: Symbol `%s' has different size in shared object, consider re-linking
/staf
Stale NFS file handle
stat reports %d blocks: %d
Streams pipe error
Structure needs cleaning
STRV
```

```
stuffing block %d
Success
Sunday
--%s VALUE
s$XYj
symbol
symbol=%s;  lookup in file=%s
/SYS
syslog: unknown facility/priority: %x
system bytes     = %10u
system search path
<:t
</t-
<}t/
<$t,
<%t
:$t'
:$t+
{ t;
{ t
t_<}
t-;]
t:;]
t       ;}$
t0@Nt
t0tv
< t1<   t
t2Wj
t";5`
@t5P
@t7=
<,t71
t7)M
@t8=
@t8P
:t9G
t       9P
t       )A
<table bgcolor=%s><tr><td>%s: %s</td></tr></table><br>
<table bgcolor=%s><tr><td>%s</td></tr></table><br>
<table bgcolor=%s><tr><td></td></tr></table><br>
<tag>--%s &lt;
<tag>--%s &lt;arg&gt;</tag> %s
<tag>--%s &lt;filename&gt;</tag> %s
<tag>--%s &lt;int&gt;</tag> %s
<tag>--%s</tag> %s
<tag>--%s</tag>        %s
<tag>%s</tag>  %s
<tag>--%s VALUE</tag>
target
target file block size: %d
target filename: %s
tDPj
test for fragmentation (returns 0 if file is fragmented)
test for slack (returns 0 if file has slack)
Text file busy
tG;E
tG<%tC
<\th
tH9]
.TH %s "%d" "%s" "%s" "%s"
Thursday
Timer expired
=t%j
@tJf
t#Jt @
tL<%tH
TMPDIR
tN;}
t(@Nt
t&Nu
```

```
tolower
Too many levels of symbolic links
Too many links
Too many open files
Too many open files in system
Too many references: cannot splice
Too many users
TOP_PAD_
tOPVj
Total (incl. mmap):
to<%tk
toupper
<:tpB
t"Pj
t$Pj
t(PSj
t+PSj
Transport endpoint is already connected
Transport endpoint is not connected
:trG
TRIM_THRESHOLD_
t:RP
TRSV
true
t)RWVP
try '--help' for help.
  trying file=%s
< t.<  t
</tt>
t'<:t#
< t;<  t7<\t3F
<tt>%s [&lt;OPTIONS&gt;]
<tt>%s</tt> invocation
Tuesday
t<VW
 t)Wj
 t#Wj
tWRj
t)WS
t*WS
TZDIR
<.u_
|,;u
:/u^
}!;u
u!+}
u2QVj
u5Pj
=u8Q)+
u/;C
UCS-2BE// UNICODEBIG//
=ucs2->INTERNAL
UCS-2// ISO-10646/UCS2/
UCS2// ISO-10646/UCS2/
UCS-2LE// ISO-10646/UCS2/
=ucs2reverse->INTERNAL
UCS-4BE// ISO-10646/UCS4/
=ucs4->INTERNAL
UCS-4// ISO-10646/UCS4/
UCS-4LE//
=ucs4le->INTERNAL
uD;s
ug+=
U^h6LU3
u       Hy
u#;M
Unable to allocate buffer
Unable to determine blocksize
Unable to determine count
unable to determine filesystem blocksize
unable to determine raw device of %s
```

```
Unable to open file: %s
unable to open raw device %s
unable to raw open %s
unable to stat fd
Unable to stat fd
Unable to stat file: %s
unable to stat raw device %s
undefined symbol:
unexpected PLT reloc type 0x
unexpected reloc type 0x
UNICODELITTLE// ISO-10646/UCS2/
Universal
Unknown error
Unknown error
unspecified
unsupported version
u @P
u,@P
u(@P
u$@P
upper
u+PSV
Usage: %s [OPTION]...
US-ASCII// ANSI_X3.4-1968//
use block-list knowledge to perform special operations on files
useless bogus option
/usr/lib/
/usr/lib/gconv
/usr/lib/gconv/gconv-modules.cache
/usr/lib/locale
/usr/lib/locale/locale-archive
/usr/share/locale
/usr/share/zoneinfo
USVW
=utf8->INTERNAL
UTF-8// ISO-10646/UTF8/
UTF8// ISO-10646/UTF8/
u. u.
u WV
U.y`
uYD?e
. v.
. V.
v3IK
Value too large for defined data type
verbose
version
, version
Vhpe
violet
Vj:P
VPSW
VQRW
VQSR
VRVS
VSj
VSj0
VS~u
VS~y
v=+U
VUUU
w6r*
w9B
WCHAR_T// INTERNAL
weak version `
Wednesday
wF;U
Where <bf>OPTIONS</bf> may include any of:
  where VALUE is one of:
white
wipe
```

```
wipeslack
wipe slack
wipe the file from the raw device
 with link time reference
WPRQ
WQRP
w;r/
write error
write output to ...
Wrong medium type
WSj0
w};u
w       ;U
WVUS
wW;u
X[^_
x       2u
x49M
x       6u
x89M
xdigit
xPPS
*~xx
xy;U
XZh`
XZSh
XZt'
yellow
%Y-%m-%d
^[yY]
z. ^.
_ZPj
ZYPj
=ZYW@P
```

Part 2 (Option 1): Perform Forensic Analysis on a System

Synopsis of Case Facts

My department maintains one "open" laboratory for teaching computer literacy and providing general access to Windows machines. Students wishing to use the lab must obtain a username and password, but the machines are not dedicated to particular users and are not backed up. A policy document is distributed to students using the lab. This document is reproduced below, with portions that would identify the particular department deleted, as per GIAC practical examination guidelines:

<<DELETED>>

<<DELETED>>

Literacy Laboratory

Computer Account Guidelines

Your account is intended for use related to the <<DELETED>> Computer Literacy Course. Failure to adhere to the following guidelines may result in denial of service or deactivation of your account.

- using any other user's account
- allowing any other user to use your account
- using this account for any work not related to this course
- unauthorized use of equipment or facilities
- malicious or destructive use of equipment
- unauthorized copying of system or application software
- attempting to circumvent system security
- any actions that interfere with the usage of the systems by others
- This account may not be used for playing games or interactive "chatting" over the internet.

This document is not to be taken as the boundaries of the law but as the spirit in which your account is used. Just because something is not listed here, does not necessarily make it acceptable.

Full Name	*«FULLNAME»*
Course	*CS «COURSE»* ***Section*** *«section»*
Your username is	*«UNAME»*
Your password is	*«PASSWD»*

In addition to the policy document, all systems are bannered to indicate that personal information should not be stored on the machines, that information contained therein should not be considered confidential and may be monitored to ensure proper administration of the departmental facilities. This investigation targets a random machine pulled from the literacy lab in order to evaluate adherence to the guidelines in the policy document and our banner. Of particular interest is the installation of unauthorized commercial software, downloading of copyrighted materials such as MP3s, and viewing of pornographic images or videos in the laboratory. Routine computer use, such as software development using authorized tools, creation of Word documents and use of the Web are considered OK.

The system was not under my control before beginning the investigation nor was the system in a known state. I am, however, aware of the general class of applications that should be installed on the machine to support coursework.

The investigation was carried out on the following system:

T40p Thinkpad with (1) 60GB 7200 rpm disk and 2GB RAM running Windows XP, with one external 80GB NTFS-formatted USB drive for additional storage. This machine also dual boots Linux.

Forensics software used: AccessData's Forensics Toolkit ("FTK") v1.43a and WinHex 11.15 Specialist Edition.

Description of System Being Analyzed

A single machine was analyzed to determine compliance with lab policies. The machine is described in detail in the following section.

Hardware

TAG # UNO212-1: Dell Optiplex GX1 with serial number BD2VK, 128MB RAM, 500MHz Pentium III with one CD/DVD-ROM and one internal 3.5" floppy drive, running Windows 2000. This machine was used in the lab primarily for computer literacy training and software development, using C and Java. A variety of C and Java development tools, including NetBeans, Microsoft Visual Studio, and the Java JDK were expected to be installed. Microsoft Office and web browsers such as IE or Mozilla were also expected.

TAG # UNO212-2: 6448.6MB (~6GB) Western Digital IDE hard drive with serial # WM6271130795, model # AC26400-75R7 (installed in machine above, imaged for investigation).

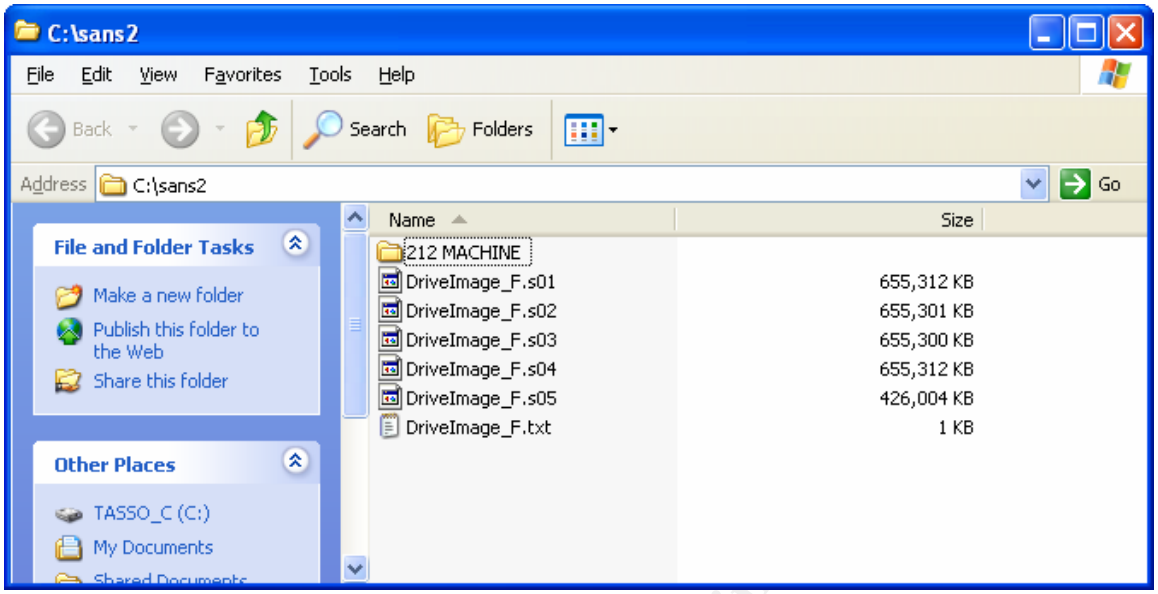
The previous evidentiary items are associated with case identifier SANSPRACTICAL-2.

The system is a 500MHz Pentium III as described above, connected to a switched 100Mb Ethernet. The machine has a connection to the Internet through a 12 X T1 backbone network.

Image Media

One hard drive, with TAG # UNO212-2 was imaged on 1/21/2004 using FTK Imager v2.0. The drive was placed in a USB drive cage before imaging. As an initial precaution, Linux was booted on the analysis machine and the target drive was attached but not mounted, in order to make modification less likely. An md5sum was run against the drive to obtain an MD5 hash. This step was performed to ensure that the FTK imager generated an umodified drive image. Then, using FTK imager, a compressed drive image was obtained, using the target drive as a source and the local drive in the T40p analysis machine as a destination (destination directory C:\SANS2). Output format was compressed SMART format, which requires less space than uncompressed dd format but is completely lossless. The FTK imager supports conversion between dd and SMART format, in case a dd image is needed for use in a different forensics analysis tool.

The MD5 hash for the image is 60D4E476A1554293216479B5D837EEC2, which matches the MD5 hash of the physical drive under Linux. The SMART image fragments have sizes as indicated below:



The size for image fragments was chosen so they could easily be backed up by burning CDs. Note that FTK processes the image fragments as though they were one monolithic image. Although in this investigation an NTFS-formatted drive was used to store the image fragments and so the size of the fragments wasn't significant (since NTFS natively supports very large files), FTK imposes a limit of 2GB for image fragments. This limitation is in place because FAT32 does not support files larger than 2GB.

The FTK image log is shown here:

Information for DriveImage_F

Frag count: 5
Frag size: 671088640

Sector count: 12578824
Sector size: 512

Bytes: 6440357888

MD5: 60D4E476A1554293216479B5D837EEC2

-end-

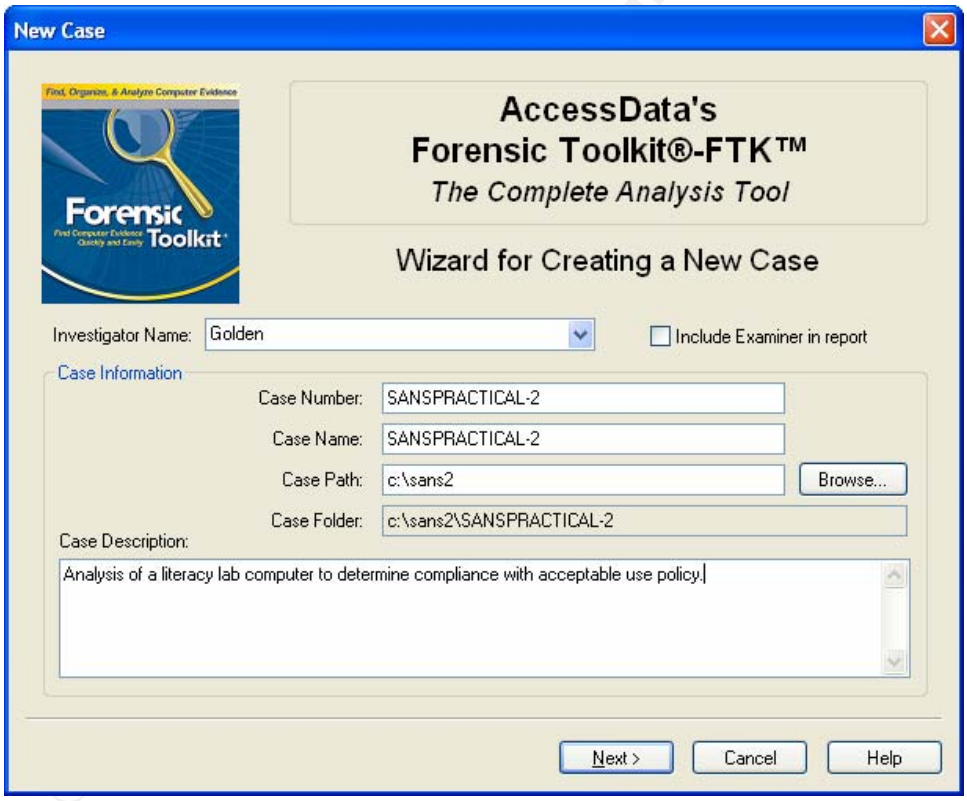
This indicates that the drive image was generated in 5 fragments, with each fragment being roughly 670K in size. The MD5 hash is the hash computed on the generated image.

Media Analysis of the System

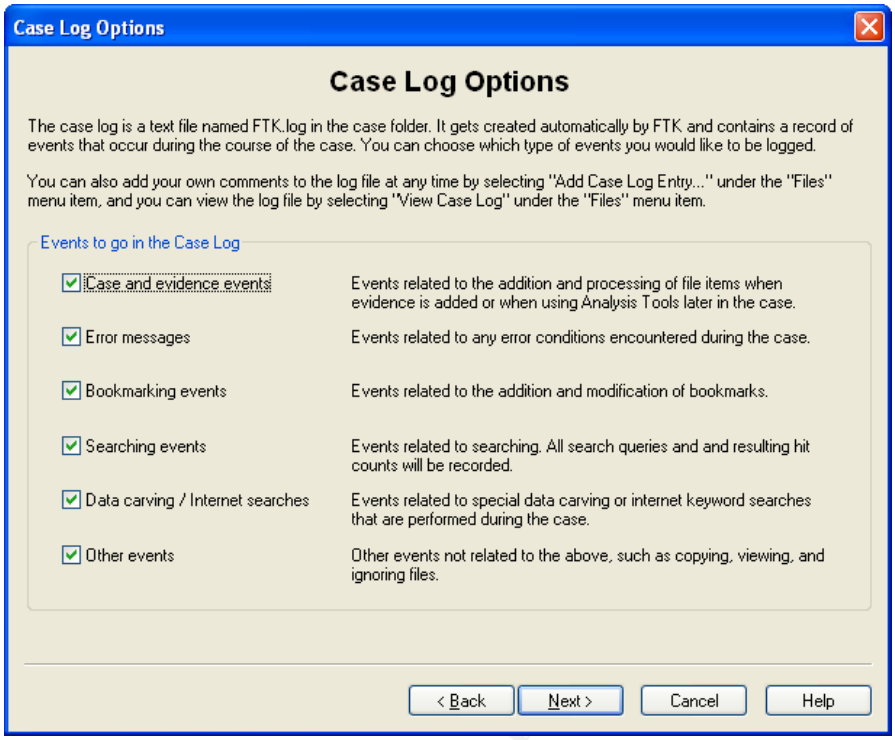
Overview

Media analysis was conducted using AccessData's FTK v1.43a. FTK is a commercial software package that has been extensively validated and operates on drive images in standard formats. It does not modify the image during investigation. FTK incorporates a set of hashes of known files (called the "known file filter", or KFF), both benign and otherwise. This set is based on Hashkeeper, maintained by the National Drug Intelligence Center [1] and on the National Software Reference Library (NSRL), maintained by the National Institute of Standards and Technology (NIST) [2]. The latest hash set available from AccessData, dated 5/19/2004, was used in the investigation.

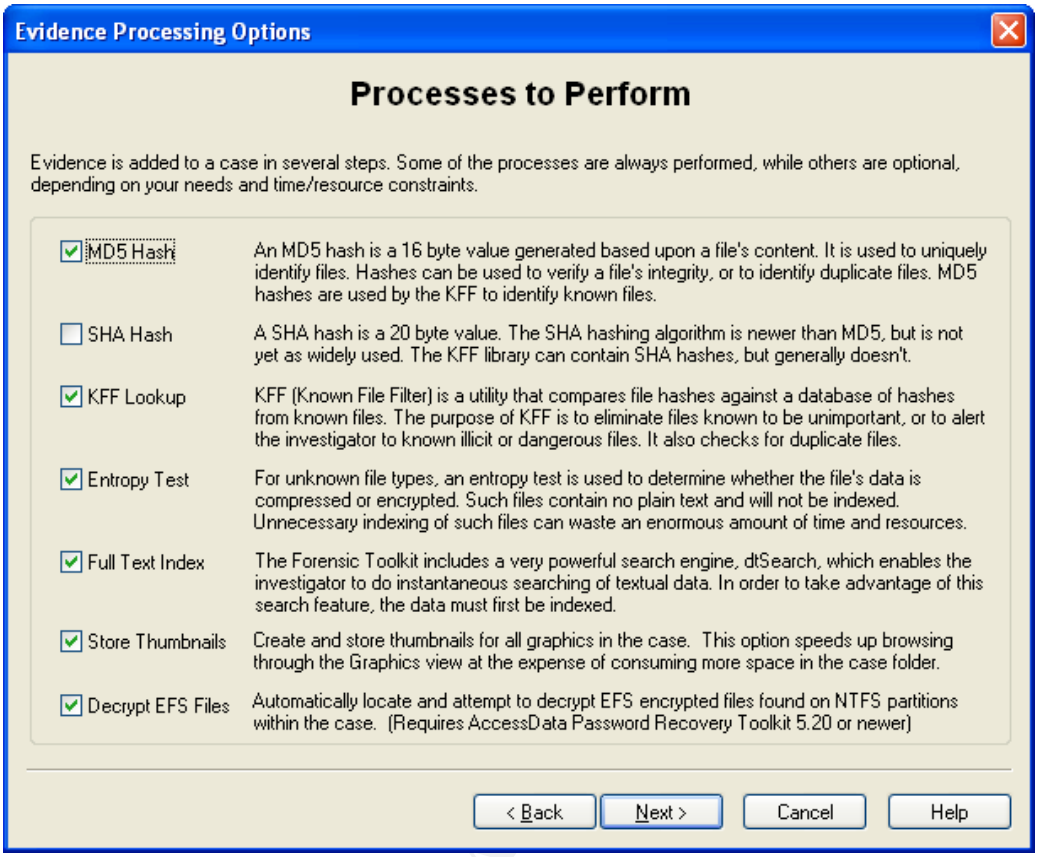
To begin, the new case wizard was used in FTK. The first screen in the wizard collects information about the new case:



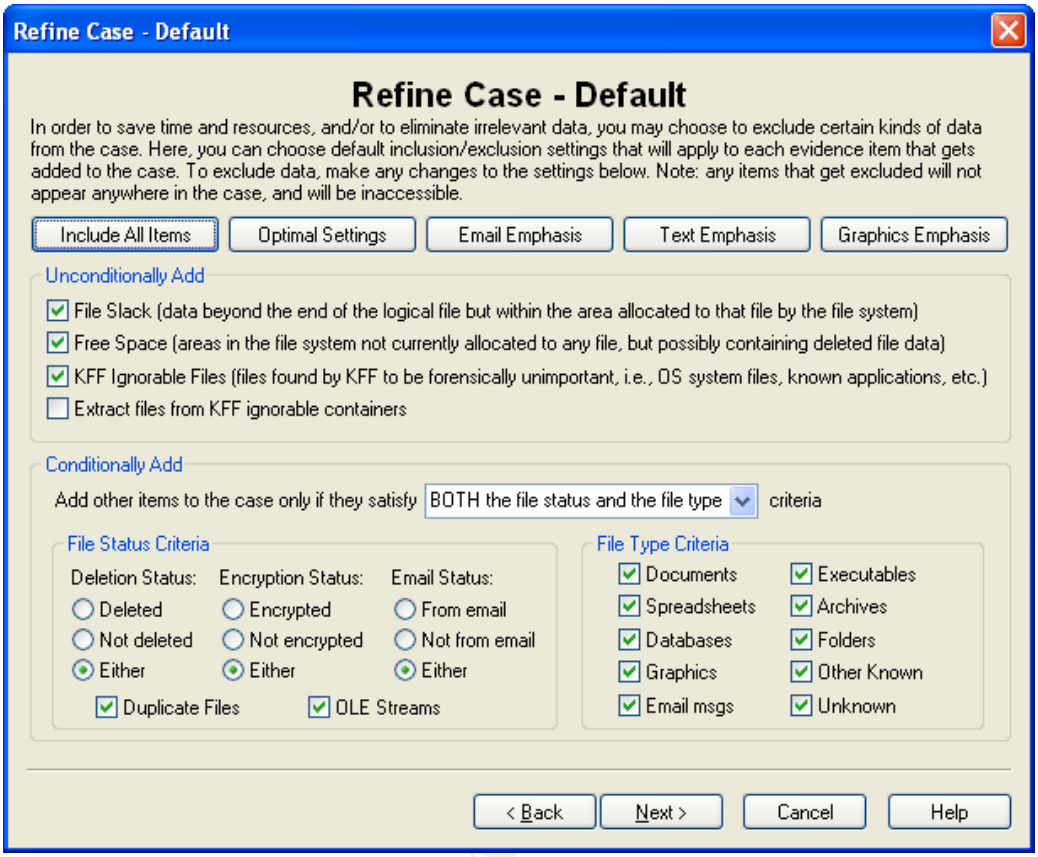
Clicking Next brings up the following screen:



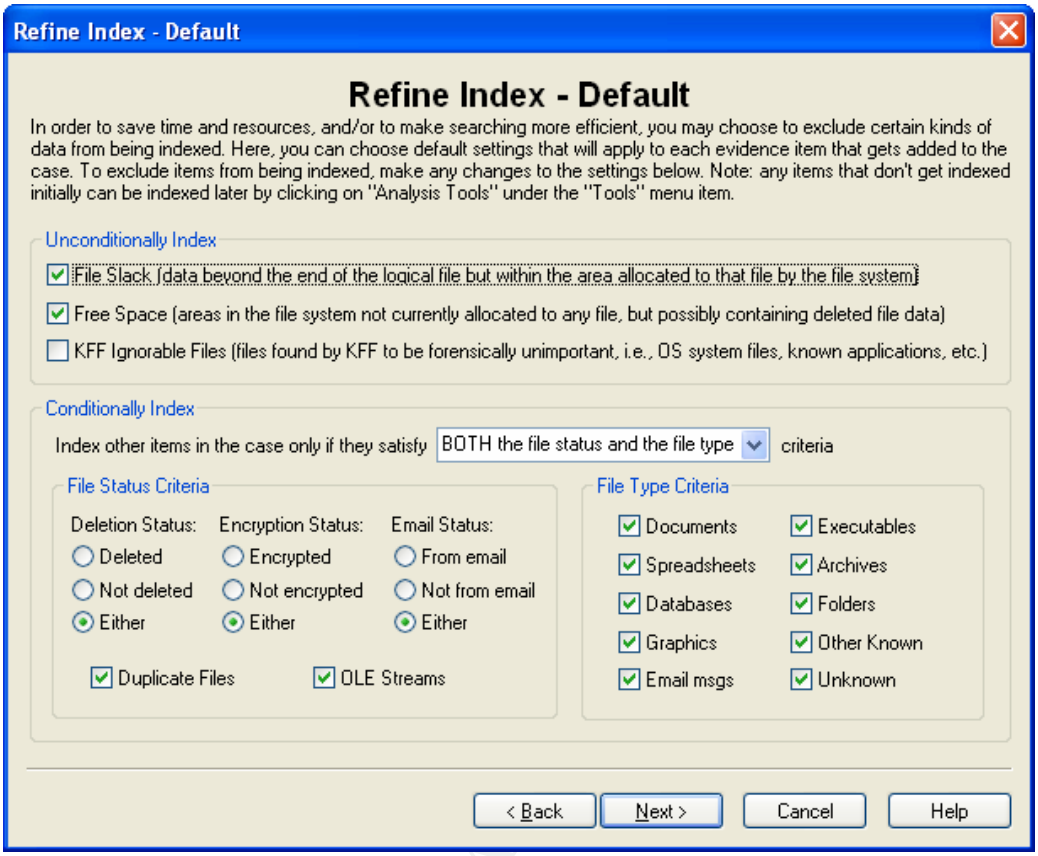
This screen allows an investigator to specify which investigatory events will be included in the FTK case log, which documents steps taken during an investigation. By leaving all of the boxes checked (the default), the FTK case log will be as detailed as possible. Clicking Next brings up the following screen:



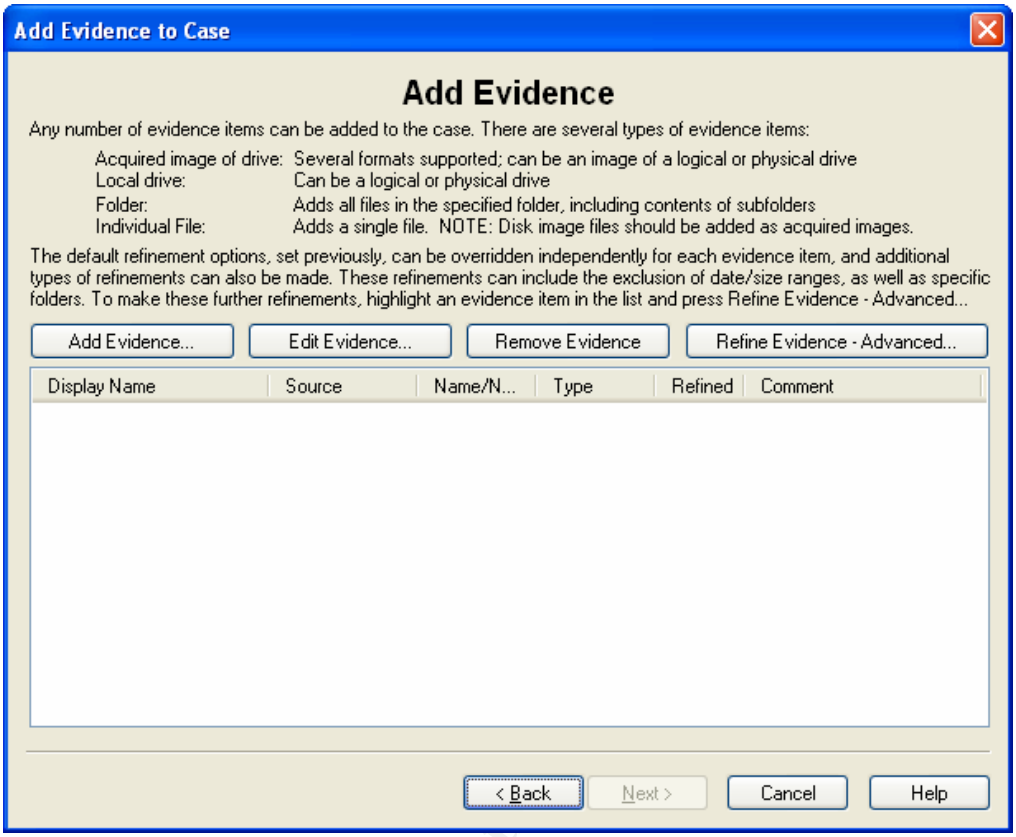
This screen allows the investigator to specify steps FTK should take in initial processing of evidentiary items. The checked options indicate that FTK should take MD5 hashes of all files, check each file against the Known File Filters (KFF) database, which allows exclusion of known files such as operating systems and application files (as well as identification of malware), that entropy tests should be performed to determine if files appear to be compressed or encrypted, that a full text index should be generated (this speeds keyword searching during the investigation), that thumbnails should be generated for all images, and that decryption should be attempted for files on encrypted NTFS partitions. Clicking Next yields:



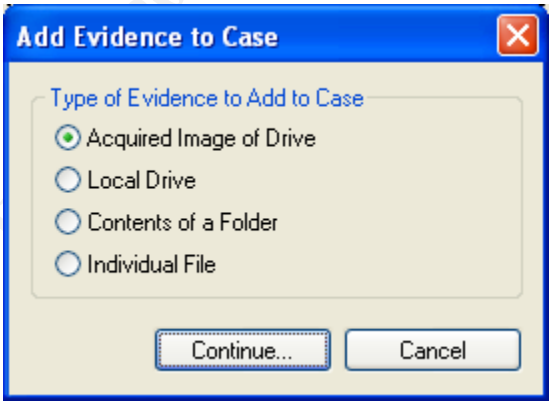
This screen allows an investigator to place emphasis on certain kinds of file evidence, such as inclusion of file slack and free space in the investigative process, whether deleted files should be processed, and which types of documents should be categorized. The default is to include file slack and free space in the investigation, to consider deleted files, and to categorize all known file types. The defaults were accepted. Pressing Next yields the following screen:



This screen is similar to the previous, in that it allows an investigator to target free space and file slack, deleted vs. undeleted files, etc., except that this screen applies to FTK's indexing process only (which builds a database of keyword locations in all evidentiary items). All defaults were accepted. Pressing Next again yields:



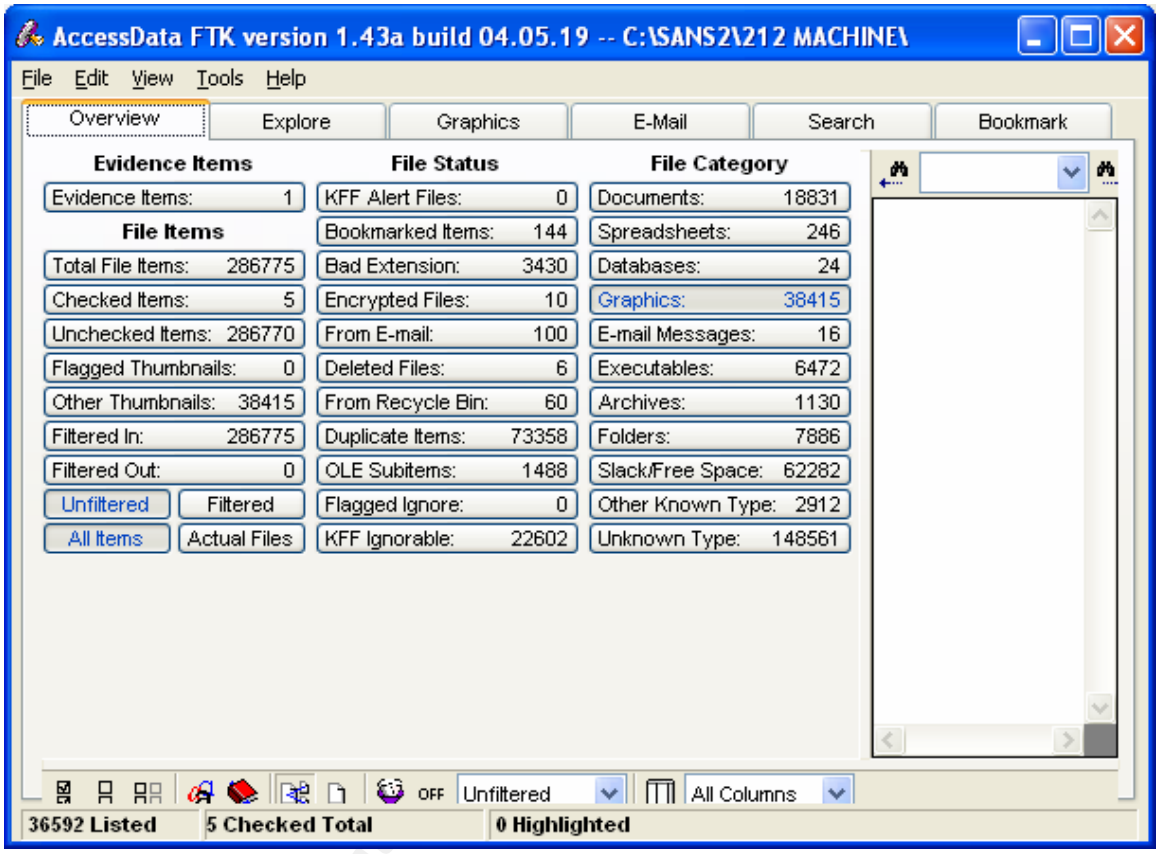
This screen allows the investigator to add (or edit, for an existing case) evidence items. By clicking Add Evidence, the following screen is displayed:



FTK can add either images, the contents of local (attached) drives, folders, or individual files to a case. The default was used, since a SMART format image of the target drive was previously captured using the FTK imager. Pressing Continue yields a file dialog box. "C:\SANS2\driveimage_F.s01" was specified as the image file (this is the first fragment of the SMART format image captured earlier—FTK automatically processes all fragments of the image). FTK then performs indexing operations, which can be very time-consuming for large drives. For the target drive in this case, approximately 2 hours

of indexing was required. After indexing, the case can be opened and the investigation can begin.

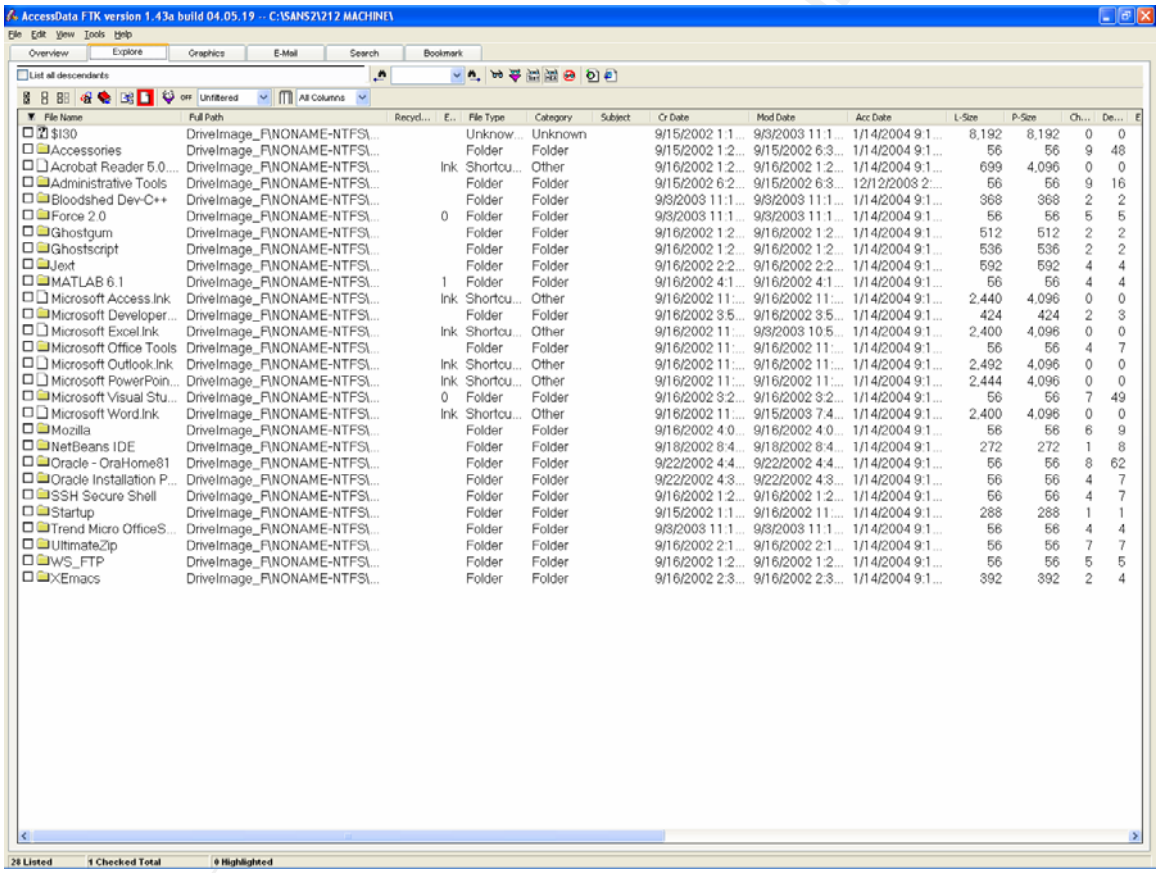
When File→Open Case is used in FTK, the case Overview is the default view. This view is depicted in the screen shot below and shows the total number of files, the number of image files, document files, databases, etc.:



Considering the FTK Overview, a total of 286,775 files were identified. Of these, FTK reports 6 deleted files, 22,602 files marked ignorable being they match hashes for known operating systems and applications files (these matched entries in the KFF database, described earlier), 73,358 duplicate files, 18,831 documents (including Microsoft Word and Excel documents), 246 spreadsheets, 38,415 images, 16 email messages, 6,472 executables, and 7,886 folders. 56 distinct users of the machine are noted by examining the "Documents and Settings" directory. Although 10 encrypted files are indicated, these were just files in the standard Microsoft Office distribution. FTK identified no contraband or potentially dangerous files which matched the latest NSRL or Hashkeeper hash sets, making existence of an existing compromise unlikely. As an added measure for detection of malware, an additional copy of the target drive was made and this drive mounted in a test system running Windows XP. A full drive scan using Norton Antivirus 2004 was performed, but no infected files were discovered.

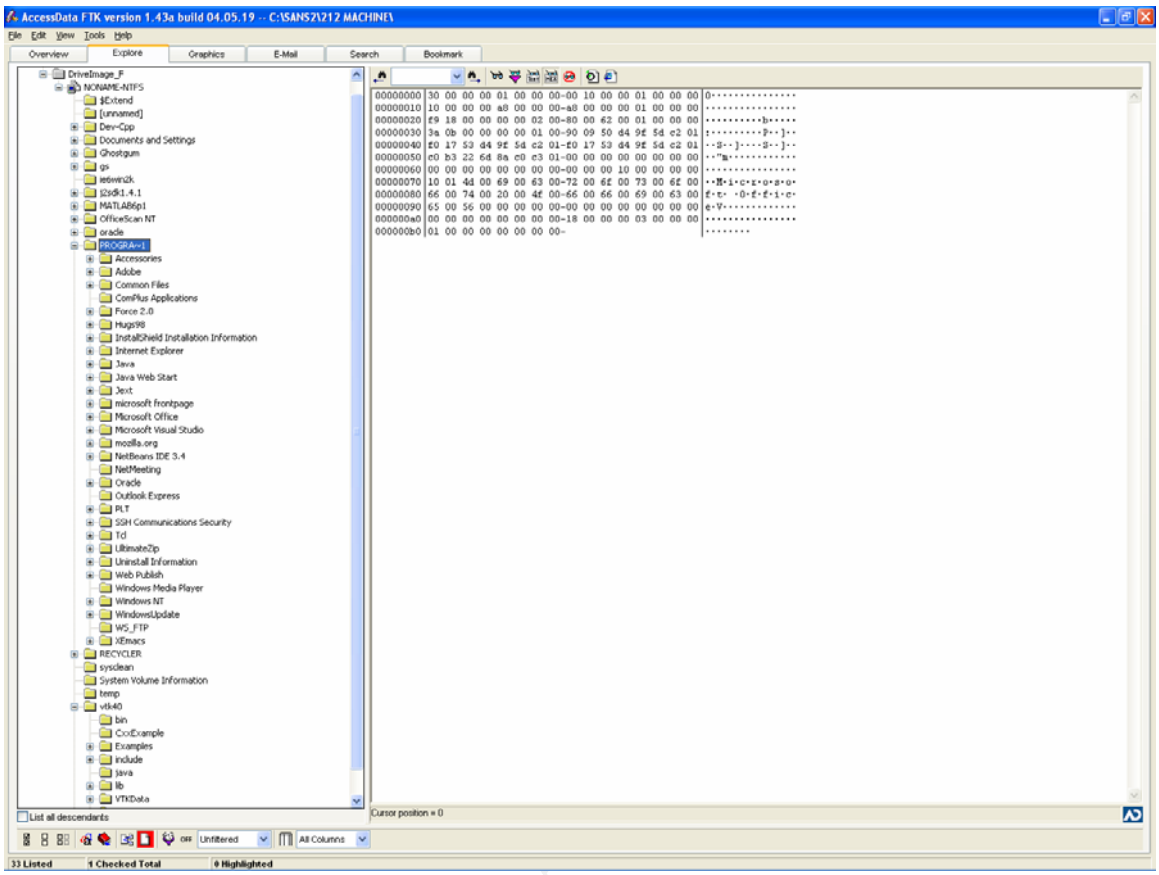
As an initial investigative step, the "Documents and Settings" subdirectory for each user was examined to note any unusual application software or startup folder settings. A description of the contents of each user's subdirectory is omitted due to the large number of users; only significant entries are noted. FTK provides an intuitive graphical interface that allows quick exploration of each of the user's subdirectories, similar to the Windows Explorer window. Evaluation of Internet Explorer history files is deferred for the moment.

The programs installed for "All Users" include the following, captured by examining the Start Menu→Programs directory. None are unusual; they include common development tools used in the laboratory, archiving tools, file transfer and remote communication utilities, and Microsoft Office.



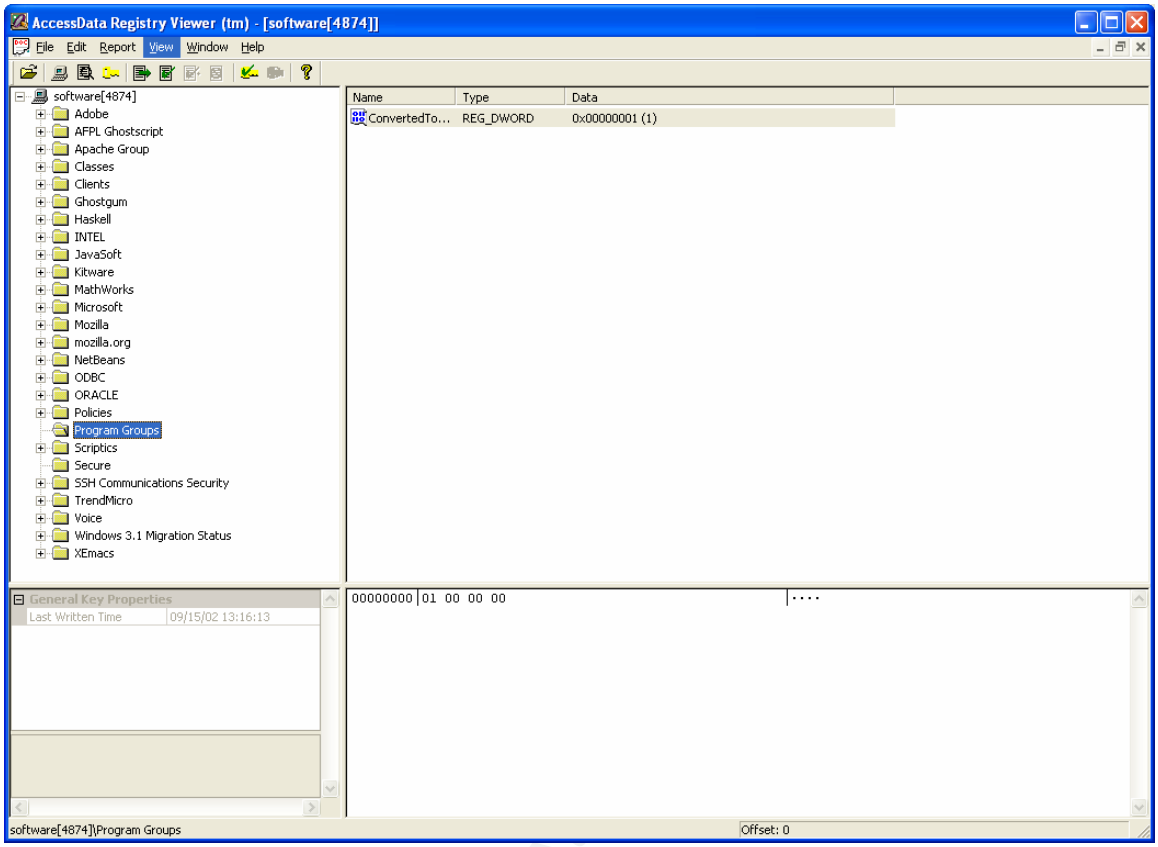
The startup folder for the Administrator account contains one entry: UltimateZip Quickstart; all other startup folders are empty.

The following folders exist in the "Program Files" directory:



All entries are reasonable. A consultation with the systems administrator revealed that the department has licenses for each of these packages (in the case of commercial software) or that the software is authorized for use (in the case of free/shareware).

The top level of the "software" file in \WINNT\System32\CONFIG, a portion of the Windows registry which holds information about installed software, revealed nothing out of the ordinary:



The “ntuser.dat” file for each user was then checked using the registry viewer in FTK. This file reveals information about recently used documents, AIM/ICQ contact lists, and Kazaa configuration information. Of particular interest are the “Software\Kazaa”, “Software\Yahoo”, “Software\Mirabilis”, and “Software\America Online” keys, which indicate installation/use of popular chat/file sharing software. None of the “ntuser.dat” files indicated installation or use of any of these packages.

The print spool directory was checked and was empty.

Use of a local email client (as opposed to using a web-based email client) was very light on the target machine. Several email messages were recovered, but all except one were class-related. One contained a comparison of several laptop computers as part of a student presentation, emailed to the instructor of the course. Others contained queries about whether an instructor had received a particular assignment. The one non-class related email discussed a Ferrari seen on the Interstate:

Message0001	
Subject:	**Ferrari Enzo (02-)**
Date:	Wed, 29 Oct 2003 09:19:51 -0600
To:	<Tracy082@aol.com>
Message Body	

This is the machine that we saw getting on the interstate. Ferrari made less than 400 of these things.
Attachment
-----Attachment2----- File name = “Ferrari Enzo (02-).jpg”

Main Message Header
++++: “DELETED To: <DELETED@aol.com> Subject: Ferrari Enzo (02-) Date: Wed, 29 Oct 2003 09:19:51 -0600 MIME-Version: 1.0 Content-Type: multipart/mixed; boundary=”----=_NextPart_000_0001_01C39DFD.CEDA8520” X-Priority: 3 X-MSMail-Priority: Normal X-Mailer: Microsoft Outlook Express 6.00.2600.0000 X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2600.0000
Sub Header
Content-Type: multipart/alte++++ive; boundary=”----=_NextPart_001_0002_01C39DFD.CEDA8520”
Sub Header
Content-Type: text/plain; charset=”iso-8859-1” Content-Transfer-Encoding: quoted-printable
Sub Header
Content-Type: text/html; charset=”iso-8859-1” Content-Transfer-Encoding: quoted-printable
Sub Header
name=”Ferrari Enzo (02-).jpg” Content-Transfer-Encoding: base64 Content-Disposition: attachment; filename=”Ferrari Enzo (02-).jpg”

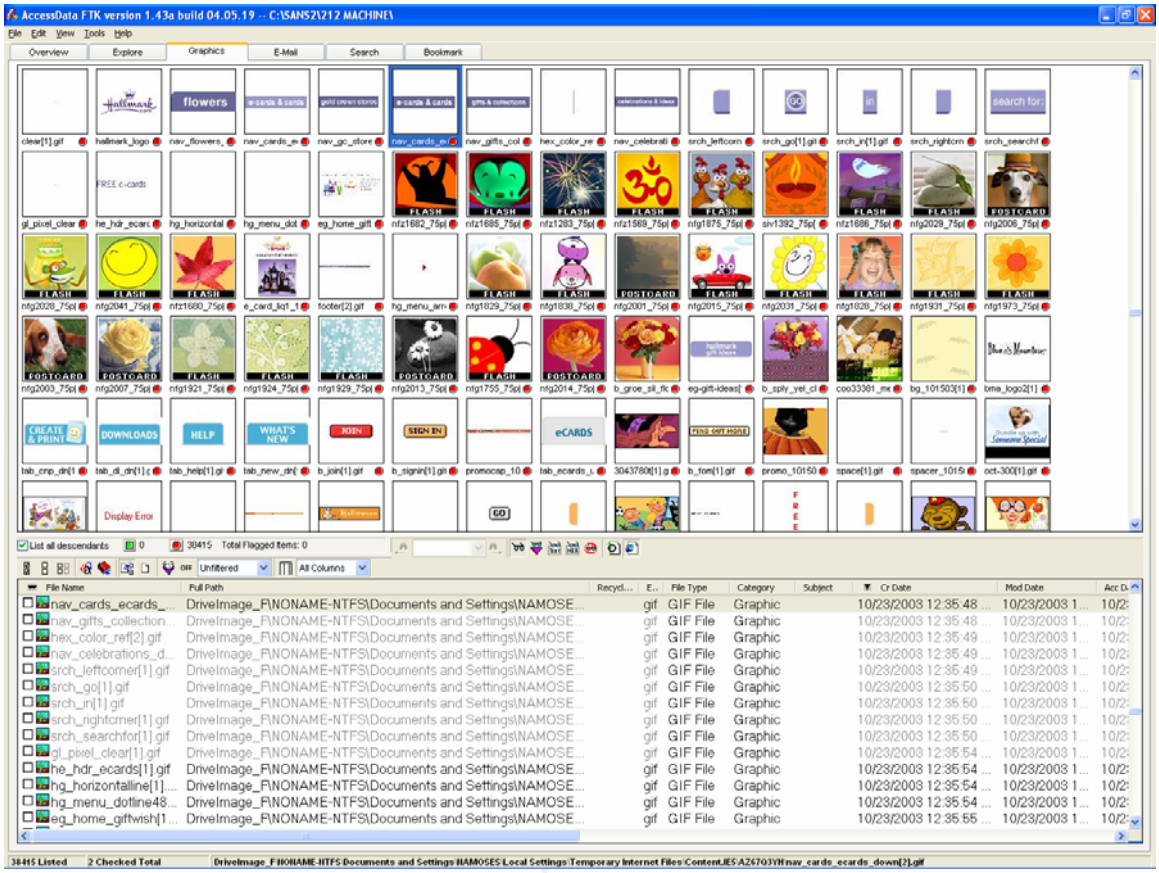
The image of the Ferrari was recovered on the machine separately, based on the filename provided in the email:

Use of the systems in the laboratory for creation of Word documents and Excel spreadsheets, for whatever purposes, are generally allowed, as many students use the lab to create resumes or to write assignments for other courses. These document types were not examined in detail.

The entire directory tree for the imaged drive was expanded and sorted by extension, in FTK. No MP3s were noted. Similarly, no AVI files other than the ones installed by Windows or by commercial applications (e.g., Microsoft Visual Studio) were noted. No MPG files were present on the computer.

The Internet Explorer history files of each user were examined briefly. Buying CDs, viewing family pictures (e.g., stored on webshots.com, a repository of user-created digital photo albums), access to Internet banking, and light usage of online games, such as those available on games.yahoo.com, by one user (name omitted to protect privacy), were noted. Discovery of more forensically interesting material, include pornographic images and illegal software, was deferred until graphics files were examined and keyword searches performed.

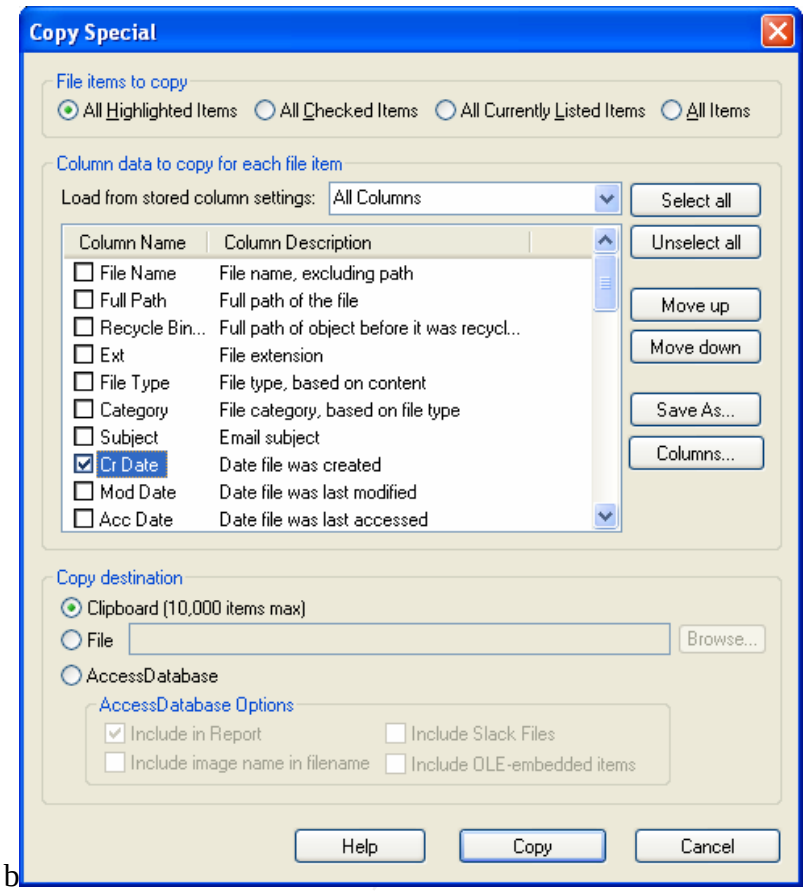
FTK reports 38,415 images total on the drive, of which 27,862 are unique. There have been several reports in the past (by students) of pornographic material being observed in the lab. The graphics thumbnail viewer in FTK was used to identify viewed pornographic content by sorting all images on the drive by creation date and then looking for groups of images that indicated a pattern of surfing for pornography. A screen shot of the graphics viewer in FTK appears below:



With the "List All Descendents" checkbox in FTK checked, all graphic images on the entire target drive can be viewed at once in the graphics viewer. Clicking on an image thumbnail highlights the appropriate entry in the file listing in the lower portion of the window, so creation, access, and modification times, as well as other file metadata, can be noted.

One user, with sanitized username "XXXOSES", browsed a large number of hardcode pornography pictures (> 100 high resolution images were found in the IE cache for this user). This clearly violates the spirit of our acceptable use policy. The user apparently made no effort to delete these images. Original images are bookmarked in FTK and are available as needed. Below, a complete list of these images with associated creation dates and MD5 hashes is provided. The listings were created by highlighting the group of image filenames in the lower portion of the FTK graphics viewer, then right clicking and copying the file info to the Windows clipboard. Any of the attributes of the selected files can be copied. A screen shot of FTK's "copy special" dialog is shown here:

b



First set of pornographic images, generated during a surfing session by the user on 10/24/2003 from 11:17am (system time) to 11:23 am:

Session # 1 (exported from FTK, all images are from directory "\Documents and Settings\XXXOSES\Local Settings\Temporary Internet Files")

```
File Name      Cr Date MD5 Hash
bbnina[1].jpg  10/24/2003 11:17:25 AM 49D79A8424BCE8CB58DF771118CC3B6E
bbtina[1].jpg  10/24/2003 11:17:25 AM BB02F2909AE053F9138A574FAA3379AD
bbjj[1].jpg    10/24/2003 11:17:26 AM F2A93D72D23ABCFA9CA3E43074C7F71D
bbmay[1].jpg   10/24/2003 11:17:27 AM 33CFB32E8826DF56C87D7DD1959668EF
bbtatiana[1].jpg      10/24/2003 11:17:29 AM 337340D91909F1145343CE28C135334F
bbjessica[1].jpg      10/24/2003 11:17:34 AM 68842BABD73B741F892F414935B9AA73
bbkate[1].jpg  10/24/2003 11:17:34 AM FE3E92C84053F0AD35E3CC900E6B37C8
bbshelia[1].jpg       10/24/2003 11:17:34 AM 05DB9F1AAEE185F5433EB4877831D6A3
bbmarie[1].jpg 10/24/2003 11:17:35 AM E4B637F77AFDD6CF06027C05F0711CB6
bbnaomi[1].jpg 10/24/2003 11:17:35 AM 5B6BEC456C37458F1AAFC956F70A250B
bbnoel[1].jpg  10/24/2003 11:17:35 AM FCE7521F37FFDFD9F06D88B62C8B8F07
bbalex[1].jpg  10/24/2003 11:17:38 AM 3A893A8D833384D24DB2EBF0D3686906
bbjenny[1].jpg 10/24/2003 11:17:39 AM A000652549DB69DCEE20DA982BBAF6E4
bbmonica[1].jpg       10/24/2003 11:17:39 AM A3B204EAE3A607572739EA6399C41625
bbisabelle[1].jpg     10/24/2003 11:17:41 AM 4CEDE5F42D1238AED8DA6F793C2C09A9
bbmelissa[1].jpg      10/24/2003 11:17:41 AM FE089F8403E0D0D042B6A8B19FF27887
bbmimi[1].jpg  10/24/2003 11:17:41 AM CC6A46DB5218B05D310275CCFD715FE2
bbsasha[1].jpg 10/24/2003 11:17:41 AM C6A324FF25C06FDEE347620C8638BBF8
bbmaritza[1].jpg      10/24/2003 11:17:43 AM 04AECE3A20A145EDEA2610E1EDFF3603
bbmary[1].jpg  10/24/2003 11:17:43 AM AC88F8EDF47A4C57683EEF78DB65B311
bbnatty[1].jpg 10/24/2003 11:17:43 AM C9B24EE405C3637A12A9DF4335921000
```

```
bbrene[1].jpg  10/24/2003 11:17:43 AM CAE9E12F8B6F99C57A958BC77A5B18D2
bbmaryjane[1].jpg      10/24/2003 11:17:45 AM E3587049159B98406425CE2A31765806
bbrebecca[1].jpg       10/24/2003 11:17:45 AM 9E3BFA05C2557F2DCD59EDF9A24078F3
bbsusan[1].jpg 10/24/2003 11:17:45 AM AA232802E43518140F948D00EC3CD348
bbchristina[1].jpg     10/24/2003 11:17:46 AM 994D6BD5782D1474DF2448BA691029B0
bbdeja[1].jpg  10/24/2003 11:17:46 AM AC3F8463DC21962482ADB27FE0CA4219
bbjulia[1].jpg 10/24/2003 11:17:46 AM DE9F9B62ED1CA4D361BFF6D366F8C63E
bblisa[1].jpg  10/24/2003 11:17:46 AM F1388E013CF23080BF19D01F5218E267
bbmargarita[1].jpg     10/24/2003 11:17:46 AM 1948D320A8BFD0381B93172FB6CB79A8
bbpamela[1].jpg        10/24/2003 11:17:46 AM BB57134C00831B9A651B2764EAD872B0
bbrain[1].jpg  10/24/2003 11:17:46 AM FBA69DB98365A2A1D46C3DC2BB78C06A
bbshannon[1].jpg       10/24/2003 11:17:46 AM 960EA6347781A1B05D0292E16F16D3BB
bbvictoria[1].jpg      10/24/2003 11:17:46 AM 347BCB94A0BF1C08806DB3B0396BD5AE
bbleslie[1].jpg        10/24/2003 11:17:47 AM 50B8AEF3830DB20880C76843400AEFB4
bbtamy[1].jpg  10/24/2003 11:17:47 AM 1A19EF313BD2952266C42F602D4BCC69
bbapril[1].jpg 10/24/2003 11:17:48 AM 8F94066EFE81F31547F86501B6AD429C
bbtracy[1].jpg 10/24/2003 11:17:48 AM 3E655133723B70D2C559F3A4B3E3B11A
bborianna[1].jpg       10/24/2003 11:17:49 AM 8F6390A2A02835EB86877B5553118771
bbbekah[1].jpg 10/24/2003 11:17:50 AM 184E193E44AB45879E91E49F32D608A8
bbcindy[1].jpg 10/24/2003 11:17:50 AM B2CE3494FF68CD6653EA0629B6AAF619
bbanna[1].jpg  10/24/2003 11:17:51 AM C90A9E0E14F804B1185BD3C83C1604F2
bbclair[1].jpg 10/24/2003 11:17:51 AM AF4AE86DB41CDBB24B989B9667BD7403
bbgloria[1].jpg        10/24/2003 11:17:51 AM 10768659265CE711E7B35E4D24E1FBD2
bbkelly[1].jpg 10/24/2003 11:17:51 AM A6A73626C26F0D1A0911D28DE617208C
bbsinstar[1].jpg       10/24/2003 11:17:51 AM A331A3BC51C32AD7AB0F28AF0738811C
bbcrissy[1].jpg        10/24/2003 11:17:52 AM 0F26BDFD68734220BAB6CA65DA84B9BB
bbkobe[1].jpg  10/24/2003 11:17:52 AM E093056BA4D1EE003350F610409AA605
0000000001_00000000000000033918[1].gif      10/24/2003            11:17:53            AM
       8B0D3F18E092A1FB0A90F96AFA2747E0
bbloni[1].jpg  10/24/2003 11:17:53 AM 556758E1013D78DBCFAACEC7E76AC2B4
bbvivian[1].jpg        10/24/2003 11:17:53 AM FDCC8909D26721CE63ACA0680D9DEFB0
text5[1].gif   10/24/2003 11:17:53 AM 43EFDAAD295808AB65C78EB66D50D2A4
230x33_2c_0825[1].gif  10/24/2003 11:18:00 AM 7E1E92AB6DB1F515FA75FE94D259A31B
bbvivian_header1[1].jpg       10/24/2003 11:18:07 AM 79EAC053E265F3F43689B02AB62A5055
bbvivian_trailer[1].jpg       10/24/2003 11:18:07 AM B81F0B0F858D463020FB9717E1B9DF35
bbvivian1[1].jpg       10/24/2003 11:18:09 AM F150F001E0162433C3024EF57FBE63EA
play_trailer_off[1].gif       10/24/2003 11:18:09 AM 686117421253535AE610FE18CA8A36AF
bbvivian2[1].jpg       10/24/2003 11:18:13 AM 0E04D1DC51FCB051DFA2E9695C481BFF
bbvivian3[1].jpg       10/24/2003 11:18:13 AM F80B166725408E8F01B107F5401EDB86
bbvivian4[1].jpg       10/24/2003 11:18:15 AM 9A3B80396B6C25C96C2B492D61395F70
bbvivian_small01[1].jpg       10/24/2003 11:18:15 AM 286EBCDC324397DFC805F184FA59FC0B
bbvivian_small02[1].jpg       10/24/2003 11:18:15 AM 1D8C6927B794EDD247F12098624F2619
bbvivian_small04[1].jpg       10/24/2003 11:18:15 AM 3AA837D8F54A60D7E10DD1142D8A696E
bbvivian_small05[1].jpg       10/24/2003 11:18:15 AM FEAF0691DF1CF5C92718379522A5A683
bbvivian_small03[1].jpg       10/24/2003 11:18:16 AM 87DAD155739C889150CBA9491676CC0B
bbvivian_small07[1].jpg       10/24/2003 11:18:16 AM 9EE9FE7C21C8B272F5892EDA2D862597
bbvivian_small08[1].jpg       10/24/2003 11:18:16 AM 2617E916C8C979564F00C03DC7D7596F
bbvivian_small06[1].jpg       10/24/2003 11:18:17 AM F3397539F4CE0F8FEE7EFB50C1618D85
bbvivian_small10[1].jpg       10/24/2003 11:18:17 AM B0075F43DF94B165901DB991E503077E
bbvivian_small11[1].jpg       10/24/2003 11:18:17 AM 19F223F94C5D17846DF7494643D63F33
bbvivian_small12[1].jpg       10/24/2003 11:18:17 AM E8AAD59192CD5E9E837C58DC627B7780
bbvivian_small09[1].jpg       10/24/2003 11:18:18 AM B8ACDC51D778E4854DDDB93CC6574584
download_off[2].gif    10/24/2003 11:18:18 AM 8C7136C8D0631C3B1A02B531500FF41B
bbkobi_header1[1].jpg  10/24/2003 11:18:39 AM 2E468DB321EE3CDC5F9D1C0F9DF4DB86
bbkobi_trailer[1].jpg  10/24/2003 11:18:39 AM FC54D8B3CFE74AC591F88CDD19A67933
bbkobi1[1].jpg 10/24/2003 11:18:41 AM EBEA62A32640E7986EBF16679BDF56CC
bbkobi2[1].jpg 10/24/2003 11:18:42 AM 8BB9BB5132073B70677E83EDD1D6ECCA
bbkobi3[1].jpg 10/24/2003 11:18:42 AM AA798F412723B2416577EE7D5C7F916F
bbkobi4[1].jpg 10/24/2003 11:18:43 AM 7249E504053B062F40EEF531CFBE9B1E
bbkobi_small01[1].jpg  10/24/2003 11:18:43 AM 8C60998AA87B7E9712612E8BCAF64E94
bbkobi_small02[1].jpg  10/24/2003 11:18:45 AM 2E28F09222BBCC48C8943019AFAE5D77
bbkobi_small03[1].jpg  10/24/2003 11:18:45 AM BCFADC129376B3883AD608BAB9AADE33
bbkobi_small04[1].jpg  10/24/2003 11:18:45 AM EEEF00EC9C596B8F09A19226644C3164
bbkobi_small05[1].jpg  10/24/2003 11:18:45 AM 50A7FC4FDF75E66E61CE8CCF89A67717
bbkobi_small06[1].jpg  10/24/2003 11:18:45 AM 2787223C7FA48939BBEE797EBEFF876D
bbkobi_small07[1].jpg  10/24/2003 11:18:46 AM 7212217582405C02A8D3D664A72CFBA2
bbkobi_small08[1].jpg  10/24/2003 11:18:47 AM 746F8C48EE3646E088CBD7E3A28FCF95
bbkobi_small10[1].jpg  10/24/2003 11:18:47 AM 3F43F586F9DA74F57BA1AD9B7E3AE705
bbkobi_small11[1].jpg  10/24/2003 11:18:47 AM E8134A894FF3362270037CEAAF8E79B7
bbkobi_small12[1].jpg  10/24/2003 11:18:47 AM CFED3ADD7F0D96CFE41E315BD473BCF6
bbkobi_small09[1].jpg  10/24/2003 11:18:48 AM 58C876FAF1E87EBFCBCA870EB3F69B58
```

```
bbcrissy_header1[1].jpg       10/24/2003 11:19:07 AM 4A5A4108CD8D81EF78196F81D9FEFC4F
bbcrissy_trailer[1].jpg       10/24/2003 11:19:07 AM DB7D8CD4689511EDAA43D12585E0EE7C
bbcrissy1[1].jpg      10/24/2003 11:19:10 AM 12E0144924EABC9CBAA0038AD19C973E
bbcrissy2[1].jpg      10/24/2003 11:19:10 AM FBF099726EC171529733B760CCE68F57
bbcrissy3[1].jpg      10/24/2003 11:19:10 AM E0FDBC432990DAC446662133D209F024
bbcrissy4[1].jpg      10/24/2003 11:19:11 AM 7B64B53AE97BD166477765C1CC4ABD0C
bbcrissy_small01[1].jpg       10/24/2003 11:19:13 AM B33CC24995167A575E13794A8E7073FB
bbcrissy_small02[1].jpg       10/24/2003 11:19:13 AM 3FE4DF9B44ACFA54BB4C0B3B012E3DBA
bbcrissy_small03[1].jpg       10/24/2003 11:19:13 AM 0B5386DBDF444902EB600DE2214AD95F
bbcrissy_small04[1].jpg       10/24/2003 11:19:13 AM 3228C46C990126A108AB3812726F5447
bbcrissy_small05[1].jpg       10/24/2003 11:19:14 AM B562992F94DB7E2418919D7B83F2FE89
bbcrissy_small06[1].jpg       10/24/2003 11:19:14 AM 4AA66E6AE7BCCEA88C49B875B63A3E8D
bbcrissy_small07[1].jpg       10/24/2003 11:19:15 AM CC0BF618FBC89914CFC748AFF2B4E193
bbcrissy_small08[1].jpg       10/24/2003 11:19:15 AM 5E7FF1E7C3E02AD81D224D7F6E296414
bbcrissy_small09[1].jpg       10/24/2003 11:19:15 AM 450F9721FCA3CF31C9EB6E1952FC2505
bbcrissy_small10[1].jpg       10/24/2003 11:19:15 AM 0EF9C9CE89204F765E14EE4028D01EE0
bbcrissy_small11[1].jpg       10/24/2003 11:19:16 AM 8181038B655A39A4E7EC33346EC27667
bbcrissy_small12[1].jpg       10/24/2003 11:19:16 AM 9C72EB82F7D7CDB85B9290EC00ADE363
bblisa_header1[1].jpg  10/24/2003 11:20:20 AM 8BD730F468CCD14F91EAB67E835FF335
bblisa_header2[1].jpg  10/24/2003 11:20:20 AM 43A82C13DC8663C40420F96447A8056E
bblisa1[1].jpg 10/24/2003 11:20:21 AM EDFA1528A8CCCA3A5AC3FA0427DA0DE3
bblisa_trailer[1].jpg  10/24/2003 11:20:21 AM 369CB1BFF221FADF99CF5F137FD89BF0
bblisa2[1].jpg 10/24/2003 11:20:22 AM 9DE74D0E74CD0C78D2B45718AF45D30C
bblisa3[1].jpg 10/24/2003 11:20:22 AM 171FA1E3E1416105DCF53DFDE79BEEFB
bblisa4[1].jpg 10/24/2003 11:20:23 AM 96E1F6D47D8D5368DD53739B6D12874D
bblisa5[1].jpg 10/24/2003 11:20:24 AM 02F53FB24355D5AE0499213A9FB93C68
bblisa_small2[1].jpg   10/24/2003 11:20:24 AM 8FAA462AFB8BFF929E4224F5F5C233A6
bblisa_small3[1].jpg   10/24/2003 11:20:24 AM 4F6688B8336A71D4BF3EAAA44A35B09F
bblisa_small10[1].jpg  10/24/2003 11:20:25 AM 2DA6D4F0C97E13FE301AD141CCF64B3B
bblisa_small1[1].jpg   10/24/2003 11:20:25 AM 32C19245792B41366DAE529ADA022827
bblisa_small4[1].jpg   10/24/2003 11:20:25 AM A044CED98524E41F2D39F1EECFEAA79D
bblisa_small5[1].jpg   10/24/2003 11:20:25 AM 134DE0E91AF9FE0A307F78E33AAE609D
bblisa_small6[1].jpg   10/24/2003 11:20:25 AM 4434A6C19A1B995A13FEC11BFC0E1865
bblisa_small8[1].jpg   10/24/2003 11:20:25 AM EBCAB9A7FD760B3A9DEBA1A1C3FA30D0
bblisa_small9[1].jpg   10/24/2003 11:20:25 AM DB3D0D3202690F3BF68938FF2C3290AE
bblisa_small11[1].jpg  10/24/2003 11:20:26 AM 12FE47107158663CCCA999C7F715BFBE
bblisa_small12[1].jpg  10/24/2003 11:20:26 AM 0D86F36327E01BF1B6B620FF1559793C
bblisa_small7[1].jpg   10/24/2003 11:20:26 AM C033B9256EF4484A30C7238ED207E918
bbbekah_header1[1].jpg 10/24/2003 11:20:40 AM 225EF23235AFA614401DED04E1FFBAF4
bbbekah_header2[1].jpg 10/24/2003 11:20:40 AM D45DC0013DA01E18FEA5A9BE69A8C8A2
bbbekah1[1].jpg        10/24/2003 11:20:44 AM 96D19DFF848C86A596316A8A1C6DC608
bbbekah2[1].jpg        10/24/2003 11:20:45 AM E9E75D8DFBFBFE60420FDCA3D00718BA
bbbekah3[1].jpg        10/24/2003 11:20:47 AM 95D9C0BA663EF341B0784978FD4CCE4F
bbbekah4[1].jpg        10/24/2003 11:20:48 AM C39500745C6E35862A3C4335A3460BCC
bbbekah_small01[1].jpg 10/24/2003 11:20:49 AM 1FFE19D844A18A1157645E89C07606DF
bbbekah_small02[1].jpg 10/24/2003 11:20:49 AM 8D388FE250EAC2BE280F40A31A330887
bbbekah_small03[1].jpg 10/24/2003 11:20:50 AM EEF7357C7094DBA8DDB883E0815599F5
bbbekah_small04[1].jpg 10/24/2003 11:20:50 AM 5DFCA9BC61ACC28DA8FAB1A5280C002B
bbbekah_small05[1].jpg 10/24/2003 11:20:50 AM 7AFA4F07BBBE470BE13634235C9AD2E7
bbbekah_small06[1].jpg 10/24/2003 11:20:50 AM EA8C56C672F2E0E03B2206CCEDDA4F14
bbbekah_small08[1].jpg 10/24/2003 11:20:50 AM 47288F004AF74103411FD9639AFA81EF
bbbekah_small07[1].jpg 10/24/2003 11:20:51 AM F9511D9D8FB4E057A1EC70E055615F4A
bbbekah_small09[1].jpg 10/24/2003 11:20:51 AM 8697DD854CE401791189AFE32F13C983
bbbekah_small10[1].jpg 10/24/2003 11:20:51 AM 2593CB8562082239D75822185999FF14
bbbekah_small12[1].jpg 10/24/2003 11:20:51 AM 76C1208E85F9CFB944FC45990733FDEB
bbbekah_small11[1].jpg 10/24/2003 11:20:52 AM 9C3FF708BE277103959CC0F924E4FC66
bbbekah_trailer[1].jpg 10/24/2003 11:20:52 AM F167677F4660CE56E03FC424850233B2
bbgloria_header1[1].jpg       10/24/2003 11:21:12 AM A1F645A28CED8F2D9F905948FDF2D30C
bbgloria_header2[1].jpg       10/24/2003 11:21:12 AM 6C12091482003CA217DCC76FA9FC4B84
bbgloria_trailer[1].jpg       10/24/2003 11:21:14 AM F6CFF7C8292D92744291CE0B3AE19D3E
bbgloria1[1].jpg      10/24/2003 11:21:16 AM 23B99F83730E0CD8EF6036FB810EF391
bbgloria2[1].jpg      10/24/2003 11:21:17 AM F862F180C42156B998D8C60092BDDCDD
bbgloria3[1].jpg      10/24/2003 11:21:17 AM 3ADAD4B5B659582551CDD774738E141B
bbgloria4[1].jpg      10/24/2003 11:21:17 AM CA1DED8997F4C061BBDFE142865B340B
bbgloria_small01[1].jpg       10/24/2003 11:21:18 AM 103F44750CEC87FF643B86E617F92175
bbgloria_small02[1].jpg       10/24/2003 11:21:19 AM 2F5124A4B78A03B0909E841CBB2F472A
bbgloria_small03[1].jpg       10/24/2003 11:21:19 AM FCA81F6C4C909B008FDAD5110520FB8A
bbgloria_small04[1].jpg       10/24/2003 11:21:20 AM F074153D7B46EB5809820AB0B88A5053
bbgloria_small05[1].jpg       10/24/2003 11:21:20 AM F3CCCEA66AFB1E5B570F97EDA5B7EDF8
bbgloria_small06[1].jpg       10/24/2003 11:21:20 AM 5D93FFE552FCE5B72451E94D09A5F1AB
bbgloria_small07[1].jpg       10/24/2003 11:21:21 AM E238B1C6C5F9D58209A00BDC58CB0991
```

```
bbgloria_small08[1].jpg       10/24/2003 11:21:21 AM 6B75DB20F4D277406288A31334BA1183
bbgloria_small09[1].jpg       10/24/2003 11:21:22 AM 06FD8CDDCCE51AFCFF8111DDCB293C8E
bbgloria_small10[1].jpg       10/24/2003 11:21:23 AM 202F7AE11DFF4D6E17BA41FB14C932C8
bbgloria_small11[1].jpg       10/24/2003 11:21:23 AM 5928A5CEAADD1F327EAC65578B0730E3
bbgloria_small12[1].jpg       10/24/2003 11:21:25 AM 94575174C0540681934B9501271349E9
bbo_moc_415x86[1].jpg  10/24/2003 11:21:50 AM 7036258258E3D301CDBEC5440EFA46D2
r7[1].jpg      10/24/2003 11:21:50 AM 192D54C71B64CA7FB7F3BE1F545CC497
banner482[1].jpg      10/24/2003 11:21:51 AM A951A0EC930AEDE1E7B2B7041559AC7E
banner463[1].jpg      10/24/2003 11:21:53 AM 80DF401599DE2B78DCD78EEE93C5238A
banner467[1].gif      10/24/2003 11:21:53 AM 0518885D723CC85DC824129826D6238B
banner486[1].jpg      10/24/2003 11:21:53 AM 55498D45110F4FBAF7191ACD6C6F7A36
hd3_on[1].gif  10/24/2003 11:22:06 AM E231B5FE8EF67D0072C2E48B29FD4518
back4[1].gif   10/24/2003 11:22:07 AM AAF15C750FE2B2D5DA7EE1722824F349
hd1_new[1].gif 10/24/2003 11:22:07 AM 3D4C9962C2FB458F79C76D86F47F0F16
hd3_off[1].gif 10/24/2003 11:22:07 AM 3F7B9FE041D0D5BA4A18AAF27872A414
hd4_off[1].gif 10/24/2003 11:22:07 AM BECC45F294AF87284D02C695F2BAF78C
hd4_on[1].gif  10/24/2003 11:22:07 AM 2F6FB7CF39AAC0CCECE4CF9C5378EA43
hd5_off[1].gif 10/24/2003 11:22:07 AM 302108049291727B65A8DEB67E87E8CF
hd5_on[1].gif  10/24/2003 11:22:07 AM 64510545BA6D8E869D82650F15E01FDE
hd6_off[1].gif 10/24/2003 11:22:07 AM F92B5111F5BE25FAEF50C4822397A65E
hd6_on[1].gif  10/24/2003 11:22:07 AM 2DEC7E234EDF350480BED59CB97D21F7
3[1].jpg       10/24/2003 11:22:09 AM 4E138A291CB028D8A325C6B0CAE8B904
hd2_new[1].gif 10/24/2003 11:22:09 AM 4CDB9A6021966213D3DA3EB672F96C20
4[1].jpg       10/24/2003 11:22:10 AM 50E33421ACE29DE487E6F4B460204483
14[1].jpg      10/24/2003 11:22:14 AM 8A9FB0EF0F9DEDD653E3B8D0E22F1ACB
26[1].jpg      10/24/2003 11:22:14 AM D5F2B881354B9809A5DA906D38356913
5[1].jpg       10/24/2003 11:22:14 AM A7763B9D616D5F4CD7840C1A9FE8CE37
bookmark[1].gif       10/24/2003 11:22:14 AM 91133153A099D50811A9CDABBB8F11A3
94652[1].jpg   10/24/2003 11:22:15 AM 9D8C772BAEDB2C4F5CF2E52F839D2C9E
94731[1].jpg   10/24/2003 11:22:15 AM 9C680A65491C8D7342E47A362949F05C
94732[1].jpg   10/24/2003 11:22:15 AM D6BAF0059682A92BAF569F35EAEFF57F
94733[1].jpg   10/24/2003 11:22:15 AM BB2A6462DC0361FB8F08577F9355E4C3
94742[1].jpg   10/24/2003 11:22:15 AM 6A24A01961ECB4C5F993258849465B79
94743[1].jpg   10/24/2003 11:22:15 AM 12C0F0D939238E8DEAD4A3CADD482FC7
94744[1].jpg   10/24/2003 11:22:15 AM 3188FEF30469D256C0750B955FB25E07
94745[1].jpg   10/24/2003 11:22:15 AM 731119D912B5C7B8F723173A302192B7
94746[1].jpg   10/24/2003 11:22:15 AM C3C3FA329B5FF4ACCAAB7A4659EEEE8F
fri[1].gif     10/24/2003 11:22:15 AM B6813502FC7C80C359BD8F4806D5A06C
trash[1].gif   10/24/2003 11:22:15 AM 856C4215304ABDE59661176CFE88B95A
92848[1].jpg   10/24/2003 11:22:16 AM 8A26791D09704FBC4837E4DCA4341DFD
92873[1].jpg   10/24/2003 11:22:16 AM 614767AE95F3EEF17CB155FCE88493BF
94308[1].jpg   10/24/2003 11:22:16 AM 9A06C1A871D14870B1F2252BC19CD593
94736[1].jpg   10/24/2003 11:22:16 AM 0D4C0D5A6B938CE9C35FB5E728BFF4AF
94737[1].jpg   10/24/2003 11:22:16 AM 5E2547327C00210745313C96D167AAE9
94741[1].jpg   10/24/2003 11:22:16 AM 9DE42A0D29253389B8AA06A160BC935D
94747[1].jpg   10/24/2003 11:22:16 AM C80005AC6985955FBA10BB41E0B0D812
94303[1].jpg   10/24/2003 11:22:17 AM 5C585633F127DF3B4C9181D7C6300DAB
94322[1].jpg   10/24/2003 11:22:17 AM CC310B79454FD72DDB5F9049C075134B
94758[1].jpg   10/24/2003 11:22:17 AM 7CB7408ABC1CE4AC1BE5FD5D9EE9F06E
94759[1].jpg   10/24/2003 11:22:17 AM 9E9654C7FE909C87A8CAB53C7683037B
94763[1].jpg   10/24/2003 11:22:17 AM 0FAA943B43C6573010BF707E268F4903
94764[1].jpg   10/24/2003 11:22:17 AM B8021973EED4D3C2BA527BFE45531D9E
94765[1].jpg   10/24/2003 11:22:17 AM 9465490680ADA6A92B43066873930711
94805[1].jpg   10/24/2003 11:22:17 AM 49A8EF5DB7A740DF0EF4B91D511E6350
94355[1].jpg   10/24/2003 11:22:18 AM 7CC49BE988335123F42D36B74D6F970B
94357[1].jpg   10/24/2003 11:22:18 AM 24380A06F9EAD78DC9537DADD1BB47EB
94362[1].jpg   10/24/2003 11:22:18 AM EC16B681118D58C7A25D188963F5F36E
94363[1].jpg   10/24/2003 11:22:18 AM 2B6E661BDF4B9B9CE98CF984048E5B13
94365[1].jpg   10/24/2003 11:22:18 AM 143216AB68992A079793F8C8B410E927
94371[1].jpg   10/24/2003 11:22:18 AM BE496E05E49CD8EC1C727E8F482D85A6
94377[1].jpg   10/24/2003 11:22:18 AM A5EADA416F5CB6230A5D74CC1B8B2449
94354[1].jpg   10/24/2003 11:22:19 AM BA7261ADE0AE093D06D76D2FBAFA93F8
94581[1].jpg   10/24/2003 11:22:19 AM 2F2BAF59B3AB6562CF096FD39FE28960
94592[1].jpg   10/24/2003 11:22:19 AM 23388E93AACF82B10EABD297E7C01D25
94598[1].jpg   10/24/2003 11:22:19 AM 1A4A7891CF4553F57D48AAD83A6E68B0
94613[1].jpg   10/24/2003 11:22:19 AM 1C5FBB4B0731330B58EDD229327C61AB
94778[1].jpg   10/24/2003 11:22:19 AM 279A8D46D7315B3118500279C7638EA7
94779[1].jpg   10/24/2003 11:22:19 AM CCE69EE8633E4976517256D744E2F335
94684[1].jpg   10/24/2003 11:22:20 AM 413B075F03B5E6B250CA819D5B8F7185
94686[1].jpg   10/24/2003 11:22:20 AM 74A199C6E8371E151BEE64923FE94550
94775[1].jpg   10/24/2003 11:22:20 AM 13791BF94DF58ECA91B17BE9A6BEC581
```

```
94770[1].jpg   10/24/2003 11:22:21 AM 1017AFE890D7712CF2EAD81CD8C84366
94688[1].jpg   10/24/2003 11:22:22 AM 805A35542B1C8ED19CEE408236FDAD20
94693[1].jpg   10/24/2003 11:22:22 AM 5F39E717F6FC8012505B8D93AD628EB5
94695[1].jpg   10/24/2003 11:22:22 AM AA967508252F872DC67C648B390C25F2
94769[1].jpg   10/24/2003 11:22:22 AM ED6F3E99FF7CC30B041A304F95333715
trade1[1].jpg  10/24/2003 11:22:22 AM C1EB108F2A5EA43304D0A25E576EB9C5
bangthumb[1].jpg      10/24/2003 11:22:23 AM FA62042C3DA72126B3331AD8CD79EBD5
mmm100[1].jpg  10/24/2003 11:22:23 AM 9846CA57F5A53D1DC055388A170F1B8B
94687[1].jpg   10/24/2003 11:22:24 AM 30B301CAE9E444C74B59C762120AF8AE
thu[1].gif     10/24/2003 11:22:29 AM 01A0456A960E271E255208039E437790
xnxx[1].jpg    10/24/2003 11:22:29 AM 62136486A9816DBB03E8159E3B72B6FC
94712[1].jpg   10/24/2003 11:22:31 AM 4611F10CC7A35FDB3F98EC4C47C9BC4D
94713[1].jpg   10/24/2003 11:22:31 AM 3F12C3F58CC17519C93968A7919036AA
94722[1].jpg   10/24/2003 11:22:31 AM EB223EB102B3929F4C903240C0494BBB
94723[1].jpg   10/24/2003 11:22:31 AM 4B9D05C7DA14E4C7935B719F2D3685A2
94714[1].jpg   10/24/2003 11:22:32 AM 8CD56C6FC52FAF626EE3A012354BF254
94715[1].jpg   10/24/2003 11:22:32 AM 9351342A92CEEFE191A9E35687B275EE
94716[1].jpg   10/24/2003 11:22:33 AM BA4BCC58DA95254718F1EF1781FB1093
94717[1].jpg   10/24/2003 11:22:33 AM 8A59D8D39E5E609A1C4BA11A58091B94
94718[1].jpg   10/24/2003 11:22:33 AM 43DFC17457FF928695A0D344F2D1F05C
94719[1].jpg   10/24/2003 11:22:33 AM 45A3D6DDE084DC7EB1CCF6D33BFA7A15
94720[1].jpg   10/24/2003 11:22:33 AM 5836B6EE7B0345451F34A06E78D6CD31
92847[1].jpg   10/24/2003 11:22:34 AM CDE393D4CF61095B8DFE123162D73376
94724[1].jpg   10/24/2003 11:22:34 AM C2DE7902E5F297CCE8C2EAA2274BF349
94725[1].jpg   10/24/2003 11:22:34 AM 558BE9B4358E94F6FCED1D90184FBEBF
94726[1].jpg   10/24/2003 11:22:34 AM 4E478C5B0613E1B9C0464663E546F9E5
92872[1].jpg   10/24/2003 11:22:35 AM 56A93F8447F197434840A42FCF4CFBE9
94024[1].jpg   10/24/2003 11:22:37 AM 7EAE91DEB04E45849A54410E5FDA5FDC
94301[1].jpg   10/24/2003 11:22:37 AM 4F1CEF09403BF5E4388510710E7A6978
94675[1].jpg   10/24/2003 11:22:37 AM E76A94DDA9CBB7A370C39D2AA69C2AB2
94727[1].jpg   10/24/2003 11:22:37 AM 752869256F15D4866CB091EFDE82A01A
94728[1].jpg   10/24/2003 11:22:37 AM 6C0C19BC167DE7D8B873EE9364670049
94734[1].jpg   10/24/2003 11:22:37 AM 679C5EB4733F945DADE44CFFC251330F
94735[1].jpg   10/24/2003 11:22:37 AM 4171BF2C427E6D3F4DDBAAC3F33F6D35
94211[1].jpg   10/24/2003 11:22:38 AM 0507E6ED6FA5D04F88A67C113C0AD914
94227[1].jpg   10/24/2003 11:22:38 AM DF719ED06A1C56F9F207582F5A4E9A72
94230[1].jpg   10/24/2003 11:22:38 AM 5F078771A43469C35254483461A7B16E
94278[1].jpg   10/24/2003 11:22:38 AM C2EAFB74FF99E9219193CA9E458E264E
94298[1].jpg   10/24/2003 11:22:38 AM 8174563B229AA98754A22388296D23BF
94557[1].jpg   10/24/2003 11:22:38 AM 56EF3FB6D80D7610F5B00E15DA0EE5B9
94567[1].jpg   10/24/2003 11:22:39 AM E96F559BD924BAE3D393D6071C212E36
94569[1].jpg   10/24/2003 11:22:40 AM 1DF51895F7D6098AAE64E01C1165A248
94582[1].jpg   10/24/2003 11:22:40 AM EA32E4D54176B4F37462F476D6E0B719
94599[1].jpg   10/24/2003 11:22:41 AM 5C601EA115816B7662AE8DD0D4035936
94601[1].jpg   10/24/2003 11:22:43 AM 869D73EA2B8537022668C6EAEA2CC308
94611[1].jpg   10/24/2003 11:22:45 AM 6C2ECB9C5A98E2884A33189A763F17FC
94615[1].jpg   10/24/2003 11:22:45 AM C342F449504F71CD259F53D71FCFC0E7
94618[1].jpg   10/24/2003 11:22:46 AM CD831D589446FFAB254A6388C5A2184B
94619[1].jpg   10/24/2003 11:22:46 AM 838CFC496303F7FB28E21D6271160C46
94620[1].jpg   10/24/2003 11:22:46 AM 372AFFF1871F9ADC3A79F85E8BE19005
94621[1].jpg   10/24/2003 11:22:46 AM 6E7DB9BD93CCF82027EEABEBDCDD4BFF
94640[1].jpg   10/24/2003 11:22:46 AM 961C3E055DA6C76054CA51ED04C15940
94664[1].jpg   10/24/2003 11:22:47 AM 85C9981E8CECEDEACD2840F9D49D5941
94671[1].jpg   10/24/2003 11:22:47 AM 78CE1B527003F2A350DFCF5996662A1D
94676[1].jpg   10/24/2003 11:22:47 AM C74971AA6D1ED18FE169EC73FA39BB62
94679[1].jpg   10/24/2003 11:22:47 AM 0B20DA7A7D948C193F33DB74265B6983
94455[1].jpg   10/24/2003 11:22:49 AM D56B52151927B6159B1C6EC2E649047C
94456[1].jpg   10/24/2003 11:22:49 AM 4197F82A88F4003F3A69AF22A63AF773
94458[1].jpg   10/24/2003 11:22:49 AM F3059E2AEE1F1EF4D0EF466D833BE603
94459[1].jpg   10/24/2003 11:22:49 AM 1B7DFBB96D1F775977A15E45DCFC3FD7
94702[1].jpg   10/24/2003 11:22:49 AM 704D82C1CEBB79EB9C7A613E5BF8FE62
94703[1].jpg   10/24/2003 11:22:49 AM 13AC08525E6A62087CBB13C2207E96B0
wed[1].gif     10/24/2003 11:22:49 AM EF349AAFF5774212A795B47B2B1D96D6
94457[1].jpg   10/24/2003 11:22:50 AM DCB55ADAC2A20BEF053062F90C1DFA18
94460[1].jpg   10/24/2003 11:22:50 AM 02B065DACE8010D515487585E2D610A9
94461[1].jpg   10/24/2003 11:22:50 AM 7B9E12DE3CB0C381AFF3A476EA385EEA
94462[1].jpg   10/24/2003 11:22:50 AM AEE11801D66BB66564500757241AAA63
94463[1].jpg   10/24/2003 11:22:50 AM 47DB3BEF641A4C237223742C8D9A3847
94464[1].jpg   10/24/2003 11:22:50 AM E64E0CD3812B6B8FE34A214C072123A9
94466[1].jpg   10/24/2003 11:22:50 AM D72C2CC678FE0F65D1F366F14AA240CB
94467[1].jpg   10/24/2003 11:22:50 AM E345E2F8286DE2873E438B5B76DC16B7
```

```
94468[1].jpg   10/24/2003 11:22:50 AM 0E6F3EE5ECA0E196DD2E9D6B72CDDF7B
94469[1].jpg   10/24/2003 11:22:50 AM C9DD0E4388A48794413F0DD6F034B762
92846[1].jpg   10/24/2003 11:22:51 AM 1B4C35F9C8B5E55063AD4F3C8C04F813
94470[1].jpg   10/24/2003 11:22:51 AM 9D64F4752D717635DBEF533D04777E59
94471[1].jpg   10/24/2003 11:22:51 AM FDA6A1B50AF4107F4F740B3ECE6CF28F
92871[1].jpg   10/24/2003 11:22:52 AM AEC9AE466ABE0B54FF6DBC269A55F72C
94493[1].jpg   10/24/2003 11:22:53 AM CE1BFE54E92B5F7996201572E1E5B375
94494[1].jpg   10/24/2003 11:22:53 AM 3826FC3C4D993C2C19C9E6898F19694A
93288[1].jpg   10/24/2003 11:22:54 AM 09DEAF037E09F4E0569319FA221F0E50
94565[1].jpg   10/24/2003 11:22:54 AM 2DCDDEFB4C0641841565569FAE226219
93291[1].jpg   10/24/2003 11:22:55 AM 17D32F05B22F23C80536FF9E4F098323
94282[1].jpg   10/24/2003 11:22:55 AM 6A5E4F5CE2DF90F40DE57C78208BE03C
94292[1].jpg   10/24/2003 11:22:55 AM 81662FA94016F2339AB6099B12EABCAB
94313[1].jpg   10/24/2003 11:22:55 AM BC080C249FF858E11DBE74E0D313A63D
94348[1].jpg   10/24/2003 11:22:55 AM F8F50FE8FBAEB71103C428F09ACD3A21
94366[1].jpg   10/24/2003 11:22:55 AM 822A2D81DC4E9D7C742AB8A12ABA6216
94374[1].jpg   10/24/2003 11:22:55 AM A8379F661838394EB680B0C44C353081
94382[1].jpg   10/24/2003 11:22:55 AM 0EA2CF1AB6E08B1342A66841F1304781
94399[1].jpg   10/24/2003 11:22:55 AM 9436CEC2358713E17744C21D88D7CE01
94426[1].jpg   10/24/2003 11:22:56 AM 5890804DC3F42EF6E080730EF6B2992F
94427[1].jpg   10/24/2003 11:22:56 AM 26610614701B2B3CA61689D5E2330ABE
94431[1].jpg   10/24/2003 11:22:57 AM 70D39DD9E88408BB91401FDE28548189
94433[1].jpg   10/24/2003 11:22:57 AM A679BE19EDBB89E92AC0086EF46B7FBE
94436[1].jpg   10/24/2003 11:22:58 AM B0D08CFB6874186AC407AC56559E878E
94437[1].jpg   10/24/2003 11:22:58 AM A5F1B2FFC859A851A9A6AA3103F5ADC5
94550[1].jpg   10/24/2003 11:22:58 AM E8ADA4A88DEB34E9E76DF3B0EABB31F1
94551[1].jpg   10/24/2003 11:22:58 AM 7CE82DFC2B0CCAAC1E96188DBE572113
94628[1].jpg   10/24/2003 11:22:58 AM 5E359411C9253BCAF4FE9B3A5596D58F
94646[1].jpg   10/24/2003 11:22:58 AM 111E45236DDE2154B92FECB87DACD7C8
94654[1].jpg   10/24/2003 11:22:58 AM 8EF2D6E99B8867DBE9B40DE1671847F7
94657[1].jpg   10/24/2003 11:22:58 AM 321B8FD3AA102D584CAC79132C14CF44
94685[1].jpg   10/24/2003 11:22:58 AM EE8D2A42AB65F2ACE055643BCDF64F41
94692[1].jpg   10/24/2003 11:22:58 AM F2FB70CE240151062D62040E6579CBAF
94699[1].jpg   10/24/2003 11:22:59 AM DDD28DD51BD57A06201658E709D51594
94701[1].jpg   10/24/2003 11:22:59 AM F718177AF82854776F81C4FF37E79B37
94708[1].jpg   10/24/2003 11:22:59 AM B94F09336ED3A76EE309A3F4E17B0385
94711[1].jpg   10/24/2003 11:22:59 AM C55DC10FD8DC82896F863820C037B53F
tradex[1].jpg  10/24/2003 11:22:59 AM 361063BCC310FEF8CB40A89D59A92AEA
bigtitsroundasses[1].jpg      10/24/2003 11:23:01 AM 991E42C5414473BFE78CAF9FC246B5F7
94681[1].jpg   10/24/2003 11:23:03 AM BE46F5258039D6AD3A54618B8CDA1EAE
elephant[1].jpg       10/24/2003 11:23:03 AM F1E4B60422DDC36200852497A5A8435D
```

Because of their explicit nature and because of the intended audience of this report, the pornographic images are NOT included. They are available on request.

Another set of pornographic images, generated during a surfing session on 10/28/2003 from 10:08am (system time) to 10:11am:

Session # 2 (exported from FTK, all images are from directory "\Documents and Settings\XXXOSES\Local Settings\Temporary Internet Files")

```
File Name     Cr Date                      MD5 Hash
bbjenny_header1[1].jpg 10/28/2003 10:08:07 AM 72E8811353138A779EF77A58E6D3D429
bbjenny_header2[1].jpg 10/28/2003 10:08:07 AM EE7155DD92EB8192E223C20BB853D0F6
bbjenny_trailer[1].jpg 10/28/2003 10:08:07 AM B790FF3FB5BE4DB8790A0E79F9EB6468
bbjenny1[1].jpg       10/28/2003 10:08:08 AM 065DB6B17B03F482B9E922C7F9ED0D37
bbjenny2[1].jpg       10/28/2003 10:08:09 AM 8A2E02439417CD7EA08E9B3436ED3B95
bbjenny3[1].jpg       10/28/2003 10:08:09 AM 162EEBCDAB8E4242BCAF57D914B6FD63
bbjenny4[1].jpg       10/28/2003 10:08:09 AM 0DCFD7756767B50101176805CCEB8365
bbjenny5[1].jpg       10/28/2003 10:08:10 AM 10C837E680178839FDE946CE4D63488E
bbjenny_small1[1].jpg 10/28/2003 10:08:10 AM 03590A831230034AC5A9EE7DA9715266
bbjenny_small2[1].jpg 10/28/2003 10:08:10 AM 7E6228C323359ABEBAA9264095DBD78A
bbjenny_small3[1].jpg 10/28/2003 10:08:10 AM 76D61A1DE986D5495E8E2A4C0A7AA311
bbjenny_small4[1].jpg 10/28/2003 10:08:10 AM C6C0FB333BB16DA45E395882BCDC94B7
bbjenny_small5[1].jpg 10/28/2003 10:08:11 AM 4B5749BD0C5CD58252B3E12371F76E26
bbjenny_small6[1].jpg 10/28/2003 10:08:12 AM A0DD3C4A0AF6AC8EFA7EB0462B11F2EE
```

```
bbjenny_text1[1].gif   10/28/2003 10:08:12 AM 22F15491F38E6AE48F3D6A77C291AF41
bbjenny_small7[1].jpg  10/28/2003 10:08:15 AM 15D5079D8A7B4841AA0FB7583B3F7A0C
bbjenny_small10[1].jpg 10/28/2003 10:08:16 AM E7B8FE48F7E3A25DFA4DD03705ED60FC
bbjenny_small11[1].jpg 10/28/2003 10:08:16 AM B304CC3587BFD907AE2832105FD72C6D
bbjenny_small9[1].jpg  10/28/2003 10:08:16 AM 0EF0C93D5CA92849E9FAD5FFE32BEA5C
bbjenny_small12[1].jpg 10/28/2003 10:08:17 AM F08A277B6312DB18E7D88A59821A90C2
bbjenny_small8[1].jpg  10/28/2003 10:08:23 AM 5B6F9559B667D51E0951654CE0A4D308
bbalex_header1[1].jpg  10/28/2003 10:09:09 AM F7FAE563F9515F7A12510E896F2F73F6
bbalex_header2[1].jpg  10/28/2003 10:09:09 AM 1D6AB31FE4ECE2360675D9F1625753E8
bbalex_trailer[1].jpg  10/28/2003 10:09:11 AM CF681F46F565E93376CF0C0DF01506AF
bbalex1[1].jpg 10/28/2003 10:09:13 AM F12427A3B16F03D0E9274B83F84A1D80
bbalex3[1].jpg 10/28/2003 10:09:13 AM 85B01F6EB32C37BE2645ECA6AB221131
bbalex2[1].jpg 10/28/2003 10:09:14 AM 8134CF169389EC9E08FF61A44EBF6E0D
bbalex4[1].jpg 10/28/2003 10:09:14 AM C4BF041950C2F21F33D55E6185DD4202
bbalex5[1].jpg 10/28/2003 10:09:14 AM 52EA6405A035B7CC655296AAF88F1A8C
bbalex_small1[1].jpg   10/28/2003 10:09:15 AM 02DB1F14FC837B55731E8D33BB717B13
bbalex_small2[1].jpg   10/28/2003 10:09:15 AM 7FCDA53A3DF003928C4D437BFB6753F1
bbalex_small3[1].jpg   10/28/2003 10:09:15 AM 5B10086D7AD0BF19DF2C52DBD92C28F6
bbalex_small4[1].jpg   10/28/2003 10:09:15 AM E2836711A26F5699527EF20B55FB8579
bbalex_small5[1].jpg   10/28/2003 10:09:15 AM E306E04D69355838B7B8E10B9D14A0FA
bbalex_small6[1].jpg   10/28/2003 10:09:15 AM 7D9EE8E16F18964D070E8BD694A34E79
bbalex_small7[1].jpg   10/28/2003 10:09:16 AM 5F0B6A65A61665DB2594E6DC593906A5
bbalex_small8[1].jpg   10/28/2003 10:09:16 AM F37F5B0329D33ADB33D44D8A86F1B48E
bbalex_text1[1].gif    10/28/2003 10:09:16 AM 3F1ECF8A3771E6EDD30435BD7180A9A6
bbalex_small10[1].jpg  10/28/2003 10:09:17 AM 0C5C5B00F5871205B4893FB1F6FDBF93
bbalex_small11[1].jpg  10/28/2003 10:09:17 AM ED368741A01274F1A74F649F7923B2AC
bbalex_small9[1].jpg   10/28/2003 10:09:17 AM 56CAAA6FD3CC9D913E3DBE5ED502886E
bbalex_small12[1].jpg  10/28/2003 10:09:20 AM 03019B062C6F2362A1553F91FDFB2D60
bbnatty1[1].jpg        10/28/2003 10:09:45 AM AAFB76A6EE378B735B593F7BD33CC775
bbnatty_trailer[1].jpg 10/28/2003 10:09:45 AM DFD3988397E562C415D0124CE29F4048
bbnatty2[1].jpg        10/28/2003 10:09:46 AM 117673737E239D00F3FF17A6ECFBA87E
bbnatty3[1].jpg        10/28/2003 10:09:46 AM D8D89C41AFFF3581FACFB6A2C97B3FE5
bbnatty4[1].jpg        10/28/2003 10:09:46 AM 512D1461178FA0BFE21909DEBCE6BB65
bbnatty_small01[1].jpg 10/28/2003 10:09:46 AM 26BE47DC2C7B5EB7828286B6C50515C2
bbnatty_small02[1].jpg 10/28/2003 10:09:46 AM A636526412C55B19A080E87EC275D09D
bbnatty_small03[1].jpg 10/28/2003 10:09:46 AM 0B14AAF1B45BE8B7C1978E27C5232D70
bbnatty_small04[1].jpg 10/28/2003 10:09:46 AM ACC67876A8061A128838E91C01A79404
bbnatty_small05[1].jpg 10/28/2003 10:09:46 AM 7413B0CC7754425A534D0DB9F96D69BC
bbnatty_small06[1].jpg 10/28/2003 10:09:46 AM ED07E9778917177D2D6D3C9F69689B48
bbnatty_small07[1].jpg 10/28/2003 10:09:47 AM AA401C7A2B1BDE223382C9EA4B9D6B8C
bbnatty_small09[1].jpg 10/28/2003 10:09:47 AM F3E8290671329184937D78C70CB71B52
bbnatty_small10[1].jpg 10/28/2003 10:09:47 AM F1B7606CA29D832A8DCB8AEB129C88B6
bbnatty_small11[1].jpg 10/28/2003 10:09:47 AM 82FFB5BDFC64BED47680ED694D56CEB7
bbnatty_small12[1].jpg 10/28/2003 10:09:47 AM 852DE35EECB3E2AA4FD845C34E3FD7C8
bbnatty_small08[1].jpg 10/28/2003 10:09:48 AM C323CDA72C6D2A23C894FBBBE27E5845
bbmimi_header2[1].jpg  10/28/2003 10:10:05 AM 5E0B30C6C74107916A3319D6C05FE742
bbmimi_trailer[1].jpg  10/28/2003 10:10:07 AM 31E488E812EF237B00A6D3A82D7CE5E8
bbmimi1[1].jpg 10/28/2003 10:10:10 AM C990155319AA3F8C28FBAFD9F86092DD
bbmimi2[1].jpg 10/28/2003 10:10:11 AM 062A12654B1C466753EF6EE990111003
bbmimi3[1].jpg 10/28/2003 10:10:11 AM 77A0E03ED657F07C8A6F2A0D5D296C16
bbmimi4[1].jpg 10/28/2003 10:10:11 AM 422F198EC01DF6E9BFE90EEF7C10EC23
bbmimi5[1].jpg 10/28/2003 10:10:11 AM 34921E548E7212247767DDBEC35536DF
bbmimi_small2[1].jpg   10/28/2003 10:10:15 AM 412D77E0A0D5A90469EF20C4E6B487B9
bbmimi_small3[1].jpg   10/28/2003 10:10:16 AM 3C12E4EC31700AF61158D6F3EB9189C9
bbmimi_small4[1].jpg   10/28/2003 10:10:17 AM 5A6FB8AFBBB8219044DB22B1369EE117
bbmimi_small5[1].jpg   10/28/2003 10:10:17 AM 44D2D4A0BDA1CF97C44F79801AEA3A81
bbmimi_small6[1].jpg   10/28/2003 10:10:17 AM E572A0C4AAEE45E09C2F129DE15FAA18
bbmimi_small7[1].jpg   10/28/2003 10:10:17 AM 96ED5B138587F7B4D20A9021C66FFC52
bbmimi_small8[1].jpg   10/28/2003 10:10:17 AM 947627386787C965A4AEA0701CC42A3E
bbmimi_small9[1].jpg   10/28/2003 10:10:17 AM 5BCCBD3A2BA2506D22C31625C81F2030
bbmimi_text1[1].gif    10/28/2003 10:10:17 AM 1ED4E9557F6F7AB3CF9D492E9D4E0386
bbmimi_header1[1].jpg  10/28/2003 10:10:18 AM 38628069C25AFDFB6750132149F0A710
bbmimi_small10[1].jpg  10/28/2003 10:10:18 AM 790E621212D5C248B808AB151D150AD2
bbmimi_small11[1].jpg  10/28/2003 10:10:18 AM E6FFE28D06F3FED590A02439F792F45C
bbmimi_small12[1].jpg  10/28/2003 10:10:18 AM 570E3FFF163B9268E2B61ECB48D0A75C
bbisabelle_header1[1].jpg     10/28/2003 10:10:26 AM E5E38D8B6FCDFE0E2B3435C8A0E82E02
bbisabelle_trailer[1].jpg     10/28/2003 10:10:26 AM 1A4234608FF14C72C2C84FF73B7EC0FB
bbisabelle1[1].jpg     10/28/2003 10:10:28 AM 3182DB696DD04098EFF0E0736F68AFA4
bbisabelle2[1].jpg     10/28/2003 10:10:29 AM 3D1FD56FA4CA3108B111610CACC59F3D
bbisabelle3[1].jpg     10/28/2003 10:10:30 AM A4726A7BB9C832D03499493F1026D055
bbisabelle4[1].jpg     10/28/2003 10:10:30 AM D01BB483C9F4D8594C6E7FA780291775
```

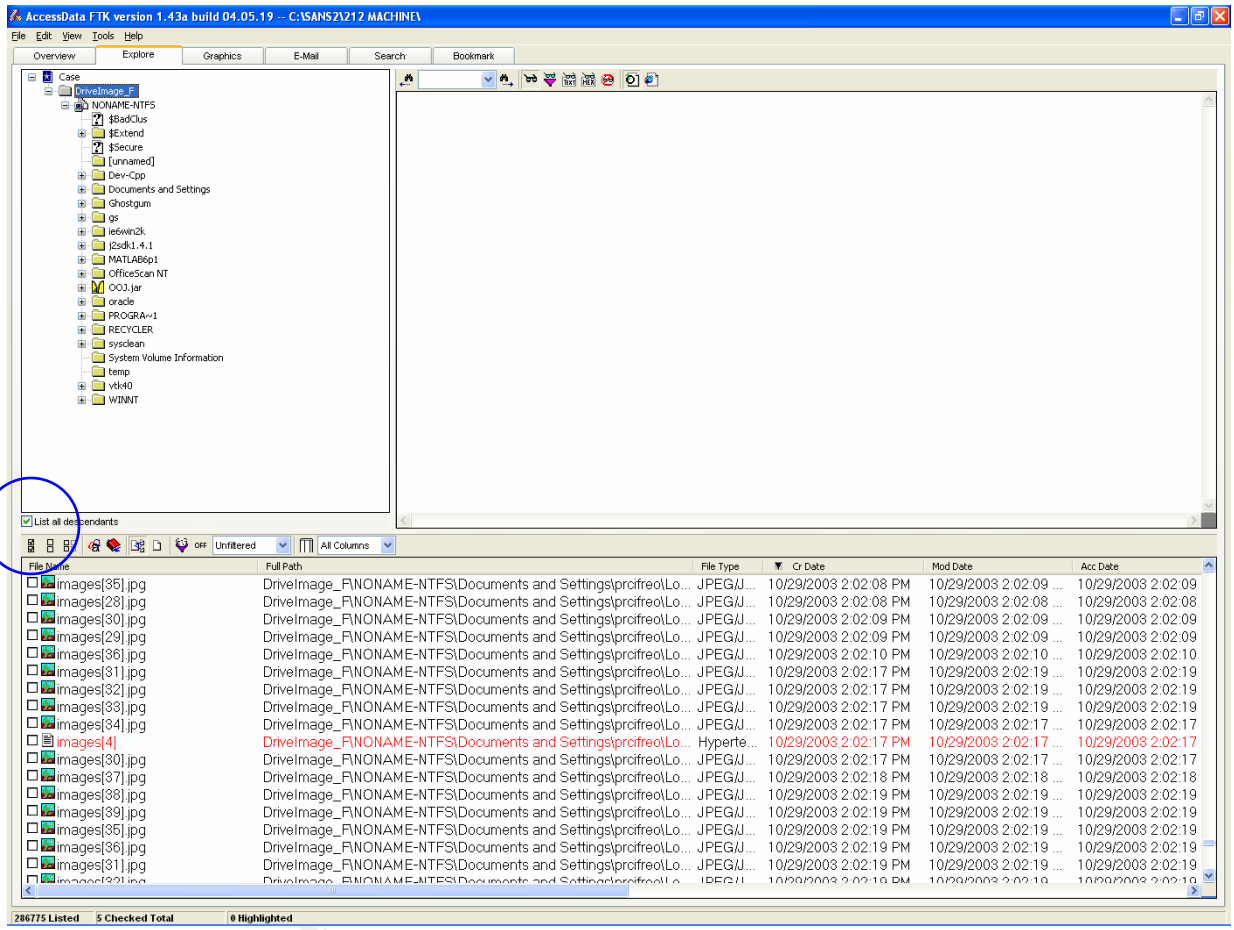
```
bbisabelle5[1].jpg    10/28/2003 10:10:31 AM AAD7ABD5390F0534527E15F32F482870
bbisabelle_small1[1].jpg      10/28/2003 10:10:34 AM E4E6A36D16BB72958E092A61F9CFEF0B
bbisabelle_small2[1].jpg      10/28/2003 10:10:35 AM B936B32B19A89A64933ABE5AA77C342B
bbisabelle_small3[1].jpg      10/28/2003 10:10:38 AM A11CF38245E558FA3399E2D5CC4086F6
bbisabelle_small4[1].jpg      10/28/2003 10:10:38 AM 52A6D27C8AEA1CF863D9BC1396D886C7
bbisabelle_small5[1].jpg      10/28/2003 10:10:38 AM A84522D18ACAC42B28353466591FA4F4
bbisabelle_small6[1].jpg      10/28/2003 10:10:39 AM C8F284AB0D63F5699EC1C6B52506A6E6
bbisabelle_small7[1].jpg      10/28/2003 10:10:39 AM C7AFDA3BF805C116152CB979B0E89CD0
bbisabelle_text1[1].gif       10/28/2003 10:10:39 AM 877820E9279240886B149653BF491F70
bbisabelle_small10[1].jpg     10/28/2003 10:10:40 AM C71D2D618400F110CCBE2F1D42DABA9D
bbisabelle_small8[1].jpg      10/28/2003 10:10:40 AM 267E50418F3504AA01E99E26926A4800
bbisabelle_small9[1].jpg      10/28/2003 10:10:40 AM 6F7A6C6BCEA6569AD3B83E395BEA013D
bbisabelle_small11[1].jpg     10/28/2003 10:10:41 AM FA2BF069294794E4357350C4102A1F33
bbisabelle_small12[1].jpg     10/28/2003 10:10:41 AM B32B7DD3F6C60F753FC7A50693F757AE
bbnina_header1[1].jpg  10/28/2003 10:10:55 AM CB84A91902337E3A8F5B5D7263235FFB
bbnina_header2[1].jpg  10/28/2003 10:10:56 AM 56A40C455F35D3FC1A2D9D230103CB7B
bbnina1[1].jpg 10/28/2003 10:10:59 AM 7678DF46B8D0141497565F7D80BA2D1E
bbnina_trailer[1].jpg  10/28/2003 10:10:59 AM ED9CBE6948CE675B58D4F95ED44D99CF
bbnina2[1].jpg 10/28/2003 10:11:01 AM E38163C12D98C0C8CE5F410F052E0600
bbnina3[1].jpg 10/28/2003 10:11:01 AM 748299F0C05F631CED566014A66AE466
bbnina4[1].jpg 10/28/2003 10:11:01 AM 367757E4BFFF32726BE04D3495B9F2F6
bbnina5[1].jpg 10/28/2003 10:11:01 AM EF969438C13BC25854A8F25BA82E434B
bbnina_small01[1].jpg  10/28/2003 10:11:03 AM D7F3D6E802C420D0FF93D6BB8324AD85
bbnina_small02[1].jpg  10/28/2003 10:11:03 AM FC9C7B9FDDB002465B7BF4E859E2DEB9
bbnina_small03[1].jpg  10/28/2003 10:11:03 AM 2FD057AB01E689F95B52C9B709E22534
bbnina_small04[1].jpg  10/28/2003 10:11:03 AM A983E52F6D30B7D7BCF2E4E0C22AE89E
bbnina_small05[1].jpg  10/28/2003 10:11:04 AM 151FA74A1C99F3E10BD1749AC33FF5D7
bbnina_small06[1].jpg  10/28/2003 10:11:05 AM 0A1E449050AF8F60A307813A383E8AC6
bbnina_small07[1].jpg  10/28/2003 10:11:07 AM 72A44A74D1373E60514F1933E40B6551
bbnina_small08[1].jpg  10/28/2003 10:11:07 AM 2C7190A588581995279076 99CF7A17CE
bbnina_small09[1].jpg  10/28/2003 10:11:08 AM 1E8A489E9AD0F73563CBE14FAFA22377
bbnina_small10[1].jpg  10/28/2003 10:11:08 AM 22514EE5F48D6FB0D43D5D9AD8AC1503
bbnina_small11[1].jpg  10/28/2003 10:11:08 AM 0B2BC34EC290D4EE08A3A1E1248C1D95
bbnina_small12[1].jpg  10/28/2003 10:11:08 AM 4BAA3BAA8F53B4C26DE6310E7683C163
bbtina_header1[1].jpg  10/28/2003 10:11:16 AM 70756086E510674CD9556F91ADAD23B0
bbtina_header2[1].jpg  10/28/2003 10:11:17 AM E9CB2CA2E2C5B5B75F85602D71CCFEA5
bbtina1[1].jpg 10/28/2003 10:11:18 AM 035483ACFFF643ED47F2E9927EA7A53C
bbtina2[1].jpg 10/28/2003 10:11:22 AM 9C4A4B221141E1F911E8B6D623802687
bbtina3[1].jpg 10/28/2003 10:11:22 AM 08ACFBDD6A3798310BD7BC79D8618CF5
bbtina4[1].jpg 10/28/2003 10:11:23 AM 2696716018291C5228C26EBB83EA4385
bbtina_small01[1].jpg  10/28/2003 10:11:23 AM 646590584B0EFF9CC1C75B9C9148CA72
bbtina_small02[1].jpg  10/28/2003 10:11:25 AM 4F0963DFB7AA7ED728662C247D7F3CD7
bbtina_small03[1].jpg  10/28/2003 10:11:27 AM 864C404FA98F3A6CFEE8FFBADE662A53
bbtina_small04[1].jpg  10/28/2003 10:11:27 AM 06A74FEF4DAF79335FE829034316C0BC
bbtina_small05[1].jpg  10/28/2003 10:11:31 AM 86897C6199B7F564049385E5A0743BC4
bbtina_small06[1].jpg  10/28/2003 10:11:36 AM DB5D0493639F23FFBA94A50A13E4A16C
bbtina_small07[1].jpg  10/28/2003 10:11:36 AM 798F9DCBD612109300AD77929E45F30C
bbtina_small08[1].jpg  10/28/2003 10:11:37 AM 6FC27CD094D5F5D58A307802F4DF380F
bbtina_small09[1].jpg  10/28/2003 10:11:40 AM E278C0E91F094A79C9ED60FC15E4E74E
bbtina_small10[1].jpg  10/28/2003 10:11:40 AM 9D2D5E942F874E0F288D7C21FEC1617C
bbtina_small11[1].jpg  10/28/2003 10:11:40 AM 2570AA9C8EF51D435581E112911772C9
bbtina_small12[1].jpg  10/28/2003 10:11:41 AM 12D72D16B43C26BD6FEC3F664AAB6505
```

Again, because of the extremely explicit nature of the images, they are not included in the report. Images are available on request.

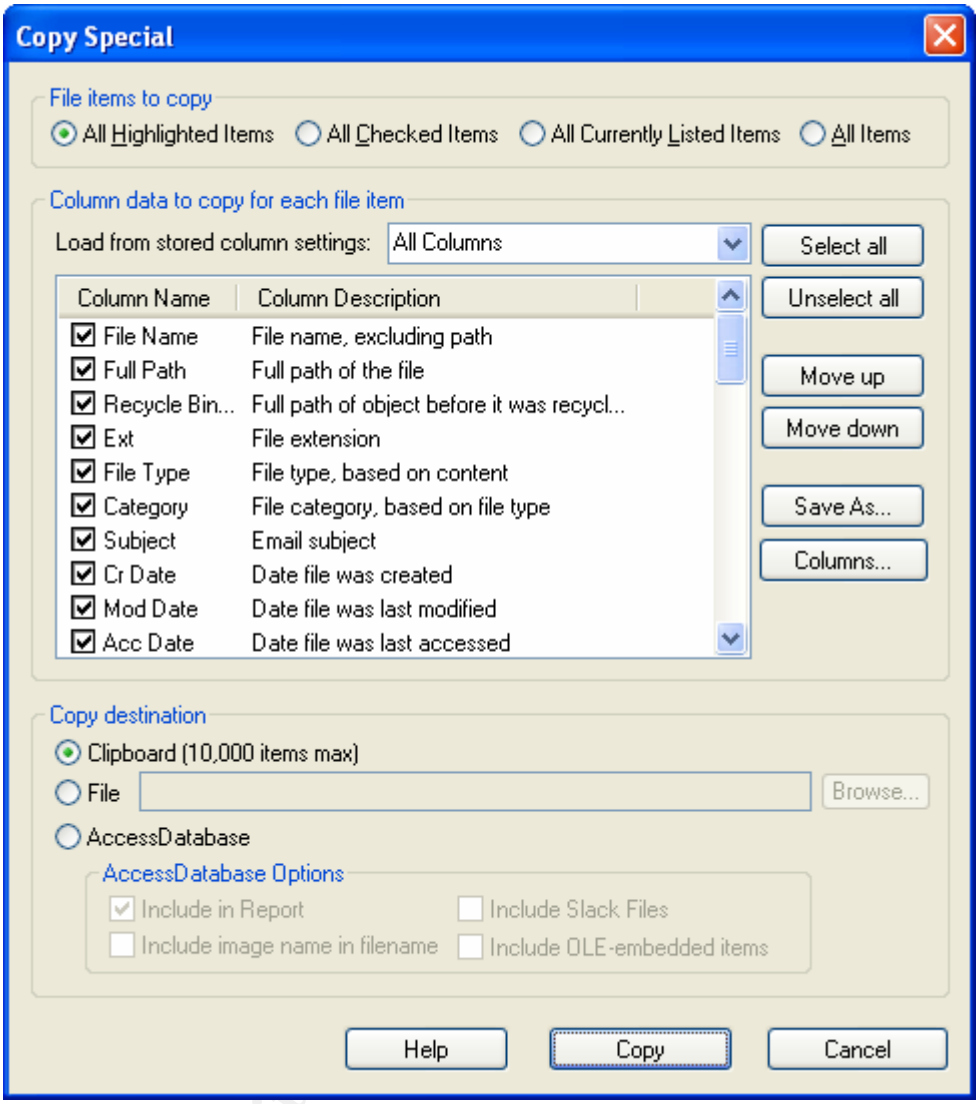
There is no evidence that other users on the machine downloaded or viewed pornographic material. Other groups of images contained anime, science fiction, collections of cellular phones (from an online shopping session), etc.

Timeline Analysis

A complete timeline of the filesystem is omitted in the body of this report due to the large number of files stored on this computer (in excess of 200,000 files). Instead, relevant portions of the timeline are included. Generating a timeline in FTK is straightforward. First, click on the Explore tab in FTK, yielding a screen like this:

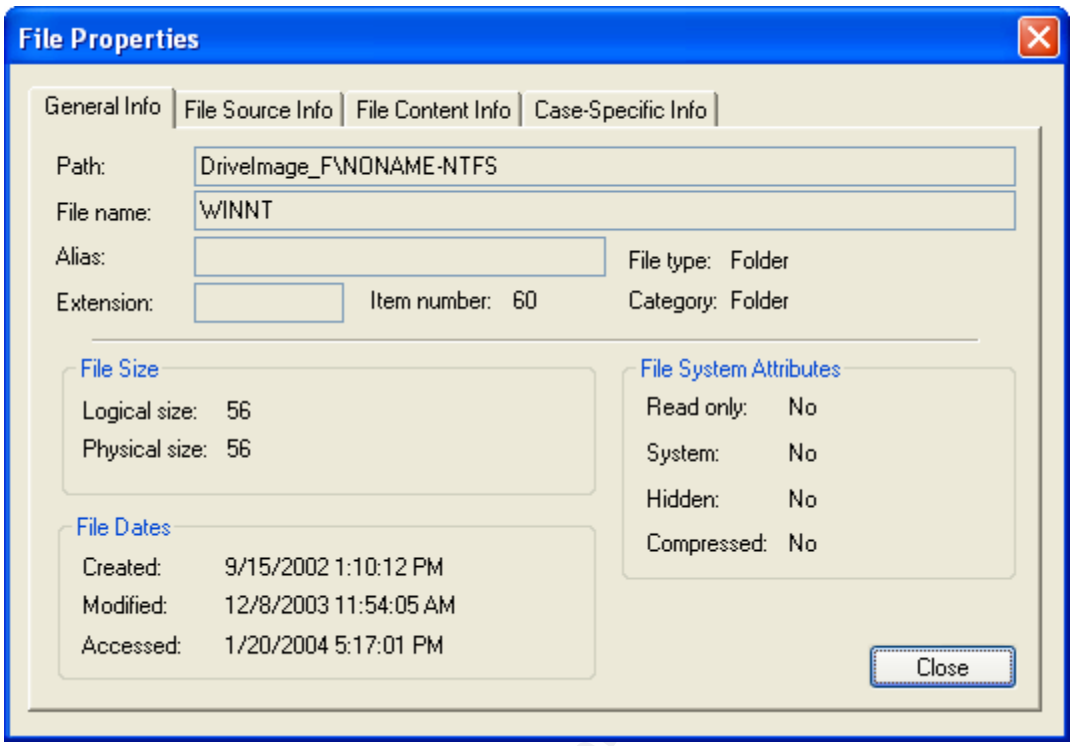


Clicking on DriveImage_F in the tree view in the left upper corner of the window selects the entire filesystem tree. Then, selecting the "List All Descendents" checkbox (circled in blue above, because it's rather hard to see) lists every file on the target drive in the bottom portion of the window. Then clicking "Cr Date", "Mod Date", or "Acc Date" will sort the file listing by creation date, modification date, or access date. The timeline can then be viewed in FTK by scrolling around in the file listing, or exported. To export the timeline, once the desired sorting criteria have been applied, right clicking on the file list in the bottom portion of the window allows the "Copy Special" dialog to be used:



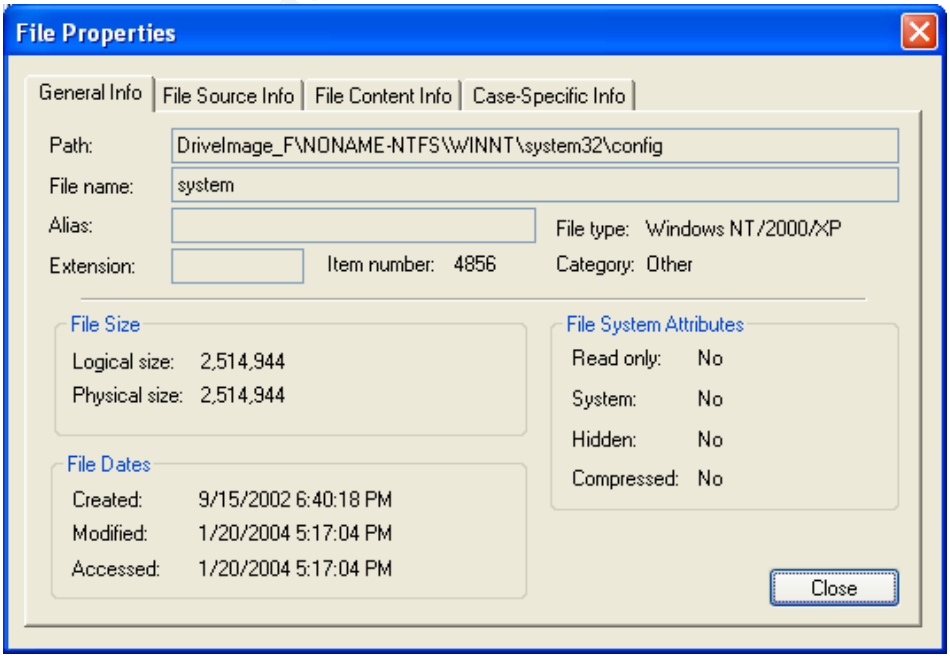
Specific columns in the file listing to be exported are specified using the checkboxes. For example, to export a timeline containing only filename and creation date columns, the corresponding checkboxes should be checked and all other checkboxes cleared. The "Copy Destination" area of the Copy Special dialog allows the data to be exported to either the clipboard, a file, or a Microsoft Access database. For the rest of the timeline analysis in this case, the dates were simply examined in FTK rather than exporting a 'textual' timeline.

The creation date of the "WINNT" directory indicates the initial install of Windows on the box since the last drive format. As seen in the following FTK screen shot, Windows was installed on this machine on 9/15/2002. Examination of creation dates of folders in the WINNT directory confirms this date.



Examination of \WINNT\ServicePackFiles indicates that the last service packs were installed 9/3/2003 at 9:39am Central.

The system was last used on 1/20/2004 at approximately 5:17pm, as shown in the following screenshot, which depicts creation, access, and modification times for \WINNT\system32\config\system, which is part of the Windows 2000 registry. This file is updated during any significant use of the computer.



The following snip of the system timeline illustrates when particular user accounts were created. These names correspond to directories in the "\Documents and Settings" folder. The first three characters of each username have been replaced with "XXX" to obscure the identity of the users.

```
File Name        Cr Date
All Users      9/15/2002 1:14:31 PM
Default User   9/15/2002 1:14:31 PM
Administrator  9/15/2002 6:42:31 PM
XXXyatt        9/3/2003 7:04:34 PM
XXXarr         9/4/2003 12:45:45 PM
XXXERRY        9/5/2003 10:10:20 AM
XXXells        9/5/2003 11:04:17 AM
XXXtter        9/6/2003 1:15:32 PM
XXXang         9/10/2003 5:19:57 PM
XXXullen       9/11/2003 10:06:27 AM
XXXrown        9/12/2003 10:11:44 AM
XXXowers       9/12/2003 11:36:20 AM
XXXam          9/15/2003 9:50:22 AM
XXXorgan       9/16/2003 5:51:30 PM
XXXeiler       9/17/2003 6:08:26 PM
XXXuidry       9/18/2003 10:52:44 AM
XXXayhi        9/18/2003 5:56:12 PM
XXXguyen       9/24/2003 5:03:43 PM
XXXldfie       9/24/2003 6:56:56 PM
XXXoss         9/25/2003 10:50:24 AM
XXXavis        9/26/2003 4:58:20 PM
XXXbdall       9/27/2003 10:20:14 AM
XXXutchi       9/27/2003 1:17:59 PM
XXXaylor       9/29/2003 6:26:06 PM
XXXuin         9/29/2003 6:39:32 PM
XXXanbor       10/1/2003 7:02:33 PM
XXXguyen       10/3/2003 10:48:50 AM
XXXizinn       10/4/2003 11:15:57 AM
XXXeBlan       10/14/2003 1:55:22 PM
XXXantlo       10/16/2003 10:42:39 AM
XXXorman       10/16/2003 12:49:15 PM
XXXifreo       10/18/2003 1:48:55 PM
XXXurns        10/20/2003 9:33:36 AM
XXXoyal        10/21/2003 10:44:38 AM
XXXega         10/23/2003 9:56:30 AM
XXXOSES        10/23/2003 12:26:25 PM
XXXoups        10/28/2003 2:27:50 PM
XXXervin       10/28/2003 6:03:31 PM
XXXonzal       10/29/2003 4:37:10 PM
XXXerrit       11/3/2003 10:55:52 AM
```

```
XXXoup          11/3/2003 3:04:08 PM
XXXarrie        11/3/2003 5:10:10 PM
XXXuth          11/4/2003 1:30:50 PM
XXXguyen        11/6/2003 3:04:49 PM
XXXeder         11/10/2003 12:44:20 PM
XXXendy         11/13/2003 10:47:25 AM
XXXelest        11/13/2003 11:50:10 AM
XXXgochu        11/13/2003 1:09:30 PM
XXXERROU        11/14/2003 12:15:27 PM
XXXee           11/15/2003 11:40:50 AM
XXXosepk        11/15/2003 1:14:12 PM
XXXulet         11/17/2003 1:25:56 PM
XXXassin        11/20/2003 12:14:37 PM
XXXolliv        11/21/2003 10:09:13 AM
XXXmith         11/21/2003 10:55:22 AM
XXXrail         11/25/2003 12:49:10 PM
XXXcemen        1/14/2004 9:15:01 AM
```

Installation of most applications in the "\Program Files" directory occurred over a short period of time following installation of Windows:

File Name	Creation Date
Common Files	9/15/2002 1:15:32 PM
Windows NT	9/15/2002 1:28:01 PM
Accessories	9/15/2002 1:28:03 PM
ComPlus Applications	9/15/2002 6:30:20 PM
Internet Explorer	9/15/2002 6:30:37 PM
Outlook Express	9/15/2002 6:30:56 PM
NetMeeting	9/15/2002 6:31:03 PM
Windows Media Player	9/15/2002 6:31:09 PM
folder.htt	9/15/2002 6:31:42 PM
desktop.ini	9/15/2002 6:31:42 PM
Microsoft frontpage	9/15/2002 6:36:03 PM
Microsoft Office	9/16/2002 11:40:56 AM
Adobe	9/16/2002 1:21:54 PM
WS_FTP	9/16/2002 1:22:39 PM
InstallShield…	9/16/2002 1:23:05 PM
SSH Communications Security	9/16/2002 1:23:05 PM
PLT	9/16/2002 1:23:29 PM
UltimateZip	9/16/2002 2:17:49 PM
Java	9/16/2002 2:20:35 PM
Java Web Start	9/16/2002 2:20:54 PM
Jext 9/16/2002 2:28:58 PM	
Xemacs	9/16/2002 2:38:24 PM
Uninstall Information	9/16/2002 2:45:11 PM
Hugs98	9/16/2002 3:19:00 PM
Microsoft Visual Studio	9/16/2002 3:21:16 PM

```
Web Publish                    9/16/2002 3:28:27 PM
mozilla.org                    9/16/2002 4:02:21 PM
NetBeans IDE 3.4               9/18/2002 8:45:21 AM
Oracle                         9/22/2002 4:26:09 PM
Tcl                            9/22/2002 5:11:30 PM
WindowsUpdate                  9/3/2003 10:07:19 AM
Force 2.0                      9/3/2003 11:14:00 AM
```

One exception is "Force 2.0", a FORTRAN 77 programming environment for Windows based on the gcc compiler. See [3] for details. This application suite was installed on 9/3/2003.

Expanding the "\Program Files" directory tree and sorting by modification date shows that no files were modified after 9/3/2003, the date that the last application ("Force 2.0") was installed.

Expanding the entire directory tree, targeting only Windows executables, and then sorting by modification time revealed that the only executables with modification dates later than 9/3/2003 were installed by the "sysclean" antivirus software and a single executable downloaded by a user XXXoup (the Google toolbar):

```
PATH: \Documents and Settings\ADLoup\Local Settings\Temporary Internet
Files\Content.IE5\MV25KBKD\GoogleToolbarInstaller[1].exe

MD5: 53E7A71599E6BF7E6A1730C8AD686FD8
```

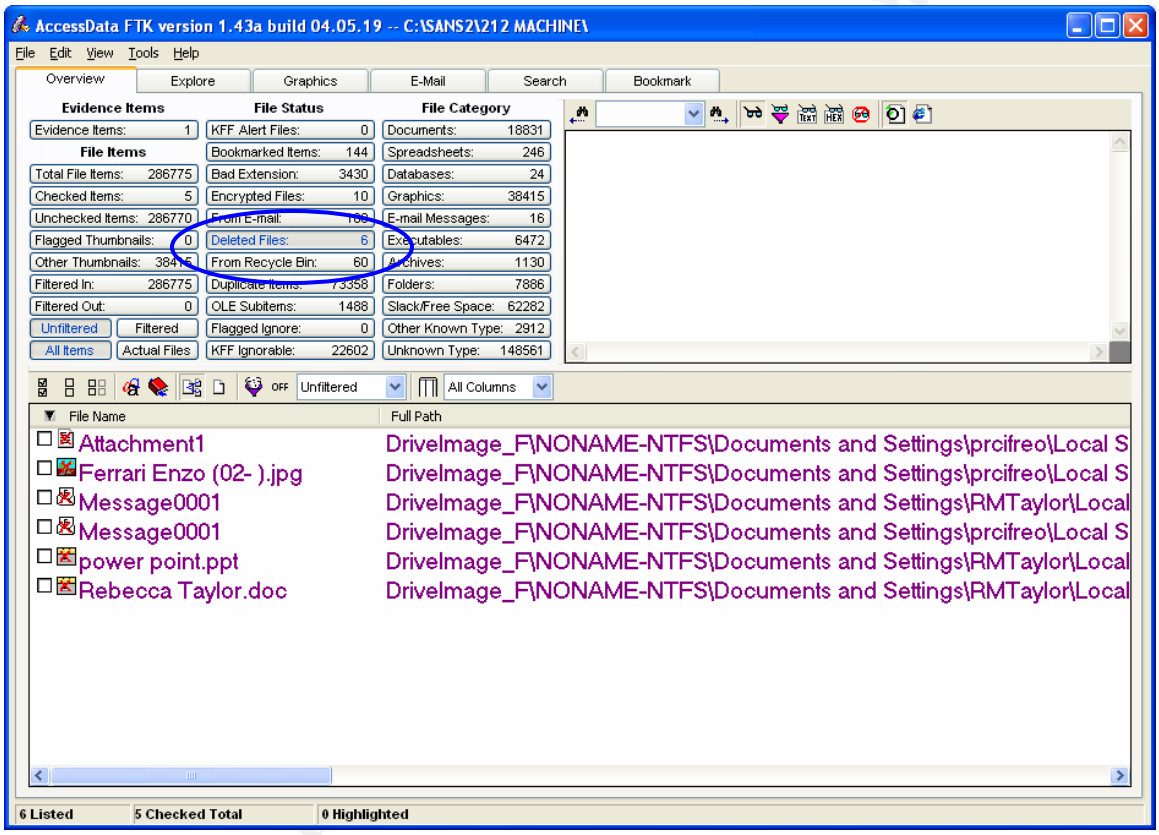
Chronology

Key events in the lifetime of the filesystem on the target machine are summarized here:

9/15/2002	→	Initial installation of Windows 2000
9/15/2002 to 9/16/2002	→	Most applications installed on machine
9/18/2002	→	NetBeans (Java development environment) installed
9/22/2002	→	Oracle, TCL installed
9/3/2003	→	Windows Update installed
9/3/2003	→	Force 2.0 (Fortran 77) installed
9/3/2003	→	Last service pack installed
9/3/2003	→	First user account created on this machine
10/24/2003	→	First surfing session involving large quantities of pornography
10/28/2003	→	Second surfing session involving large quantities of pornography
1/14/2004	→	Last user account created on this machine
1/21/2004	→	Machine taken down for imaging and analysis

Recover Deleted Files

Very few deleted files were discovered on the machine. No additional steps need to be performed in FTK to recover deleted files. Any deleted files that are completely recoverable are recovered during FTK's initial processing of the target drive image, as discussed in the Media Analysis section. All that is required to examine deleted files in FTK is to click the "Deleted Files" tab in the case Overview display:



FTK reports only 6 deleted files, none of which are significant. The first three files discuss the Ferrari Enzo seen on the interstate (see the Media Analysis section for details). The fourth file is a lab submission to an instructor. The Powerpoint and Word documents (files # 5 and 6) are corrupted and not viewable. Viewing these files in ASCII/hex view reveals nothing of interest. 60 files are in the recycle bin, but most of these are older versions of homework assignments. The "INFO2" files which track recycled files for each user under Windows 2000 were examined, but no discrepancies were found—all files listed in the INFO2 files were recoverable directly through FTK. Examining the recycle bins for users is straightforward in FTK—referring to the screenshot above, the tab below "Deleted Files" is labeled "From Recycle Bin". Pressing this tab provides a listing of all recycle bins for all users, including the INFO/INFO2 index files.

To discover other deleted material, file carving was employed. File carving searches for files in unallocated space by matching headers for known file types to identify the beginning of a file (and sometimes footers, to identify the end of a file). For example, all Word documents begin with the hexadecimal string 0xD0CF11E0. Because the structure of a file carved from freespace is lost (that is, the "file" is really just a block of data, not a proper member of the filesystem), file carving software typically locates a matching header and then assumes that the following *n* contiguous bytes on the drive are part of the file. For most carving software, the value of *n* is fixed by the investigator before file carving begins. Files that are fragmented are not easily recoverable using carving, because only the portion contiguous with the header is easily located. Still, even the initial portion of a recovered file may have evidentiary value. Similarly, files whose initial block has been overwritten are very difficult to retrieve, since no matching header will be located. String searches may still yield partial recovery for "textual" file types, though. Because files carved from freespace are not participants in the formal filesystem on a partition, no metadata is typically available. This means that filenames, creation times, modification times, access times, etc. are not recoverable, because this metadata is typically maintained by the filesystem and **not** stored in the data area of a file. File carving software simply makes up unique names for discovered files.

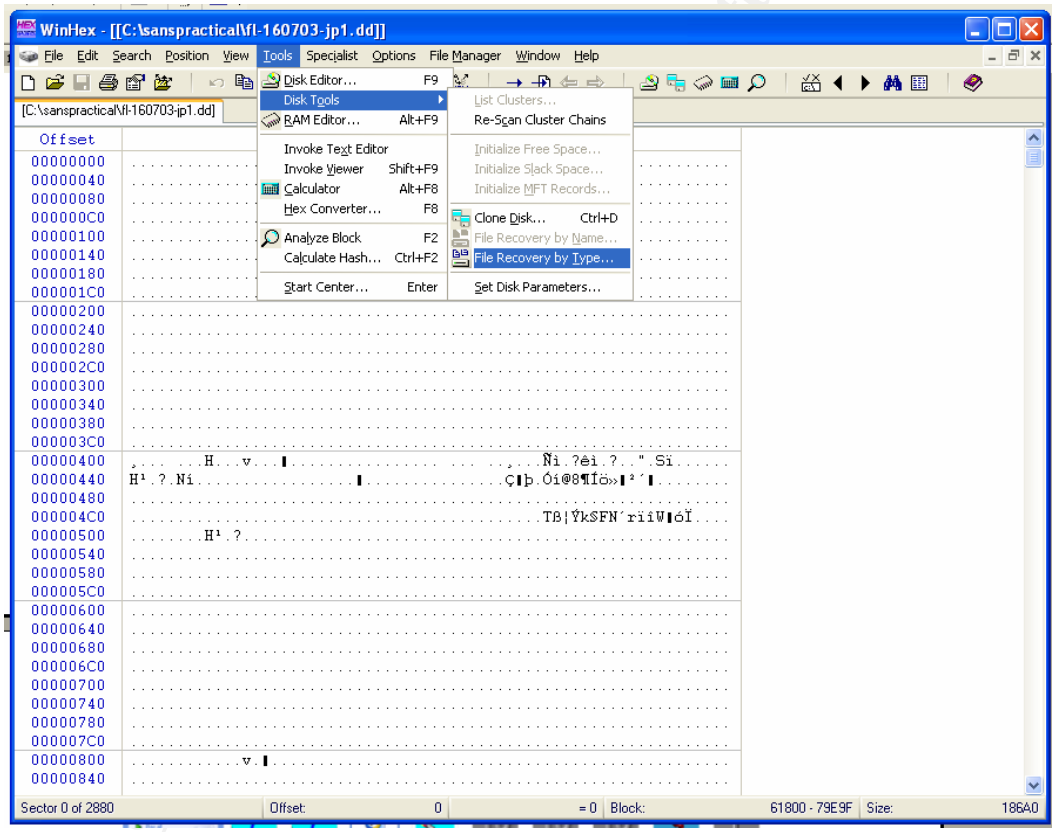
Because FTK file carving is inflexible (covering only a few file types, with no user-definable carving possible), WinHex 1.15 was used on a copy of the drive image. A dd format image was created from the SMART format target drive image using the FTK imager (the FTK imager was discussed in Section Image Media).

WinHex file carving covers the following types, with other types being user-specifiable:

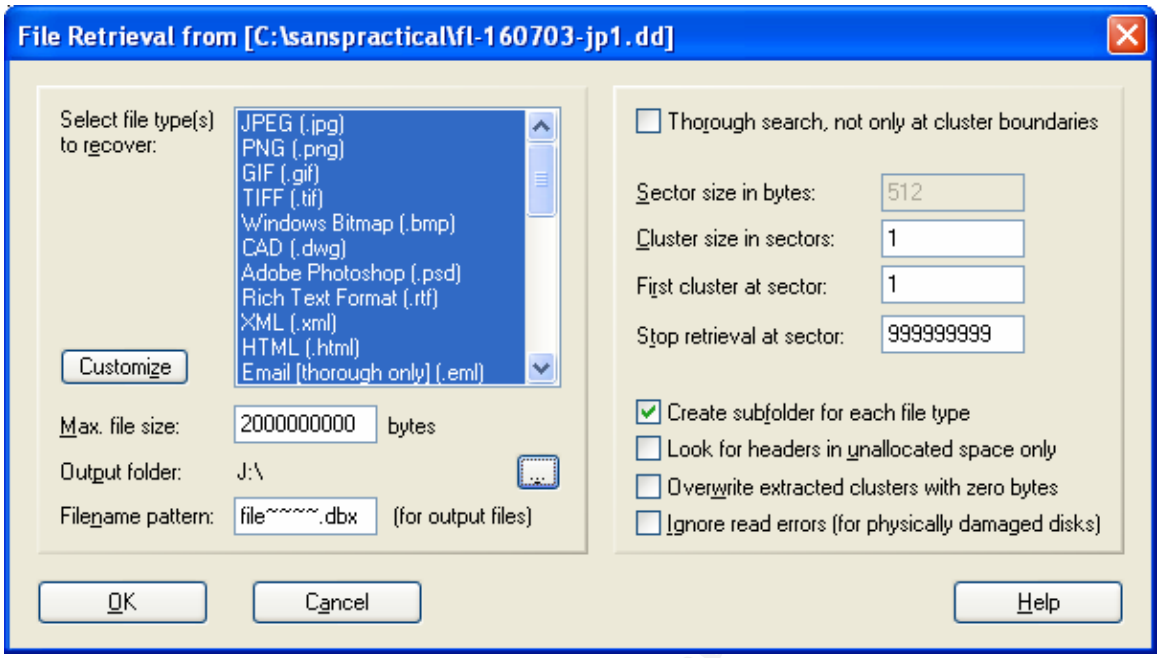
JPEG
PNG
GIF
TIFF
Windows Bitmap
CAD
Adobe Photoshop
Rich Text Format
XML xml
HTML html
Outlook Express
Outlook
MS Word/Excel
MS Access
WordPerfect
PostScript
Adobe Acrobat
Quicken
Windows Password

ZIP Archive
RAR Archive
Wave
AVI
Real Audio
Real Media
MPEG mpg
Quicktime
Windows Media
MIDI

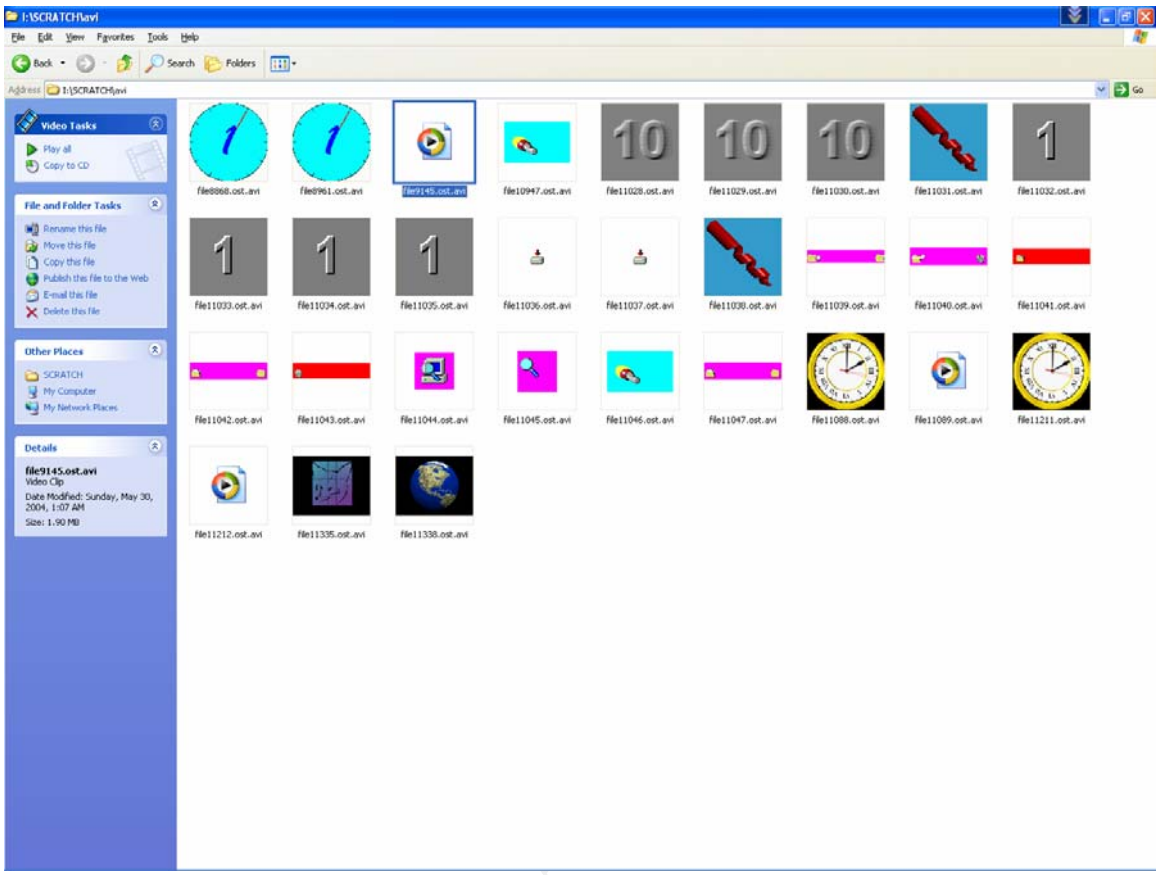
File carving operations are available under the Tools → Disk Tools → File Recovery by Type menu in WinHex:



The file carving dialog allows a number of custom settings, shown in the screenshot below:



A maximum file carving size of 2,000,000 bytes was used, with output directed to an 80GB USB drive (drive J:, as shown in the dialog). All supported file types were selected. Once carving was complete, 16,803 files were carved from free space in the image. The vast majority were .BMP, .GIF, .JPG, and .PNG image files. None of the carved 5 .MPG files were playable. None of the 30 carved .AVI files were significant, though most were playable. Thumbnail view of the recovered .AVI files is provided below:



File carving yielded no recovered MP3s. As discussed previously, no undeleted MP3s were discovered on the machine. Further, there is no evidence that file sharing software was installed on the machine.

753 ZIP files were carved. A script was used to execute the command "unzip –v" on each of the files, to determine if each ZIP file was usable. Only 5 were intact—all of these contained collections of Java class files (note: the files may have actually been Java JAR files, since ZIP and JAR files share a common header).

3 carved high-resolution (> 200x200) JPEGs could be considered "inappropriate" for viewing in the laboratory out of 2,607 JPEG files retrieved. These are omitted because they still exceed PG-13 ratings, but are available on request. No hardcore pornography was noted, providing further evidence that the machine was probably used for viewing of hardcore material by only one user, XXXOSES, as noted previously.

None of the 8,943 carved GIF files had any obvious forensic significance. All were buttons, banners, etc. for web pages.

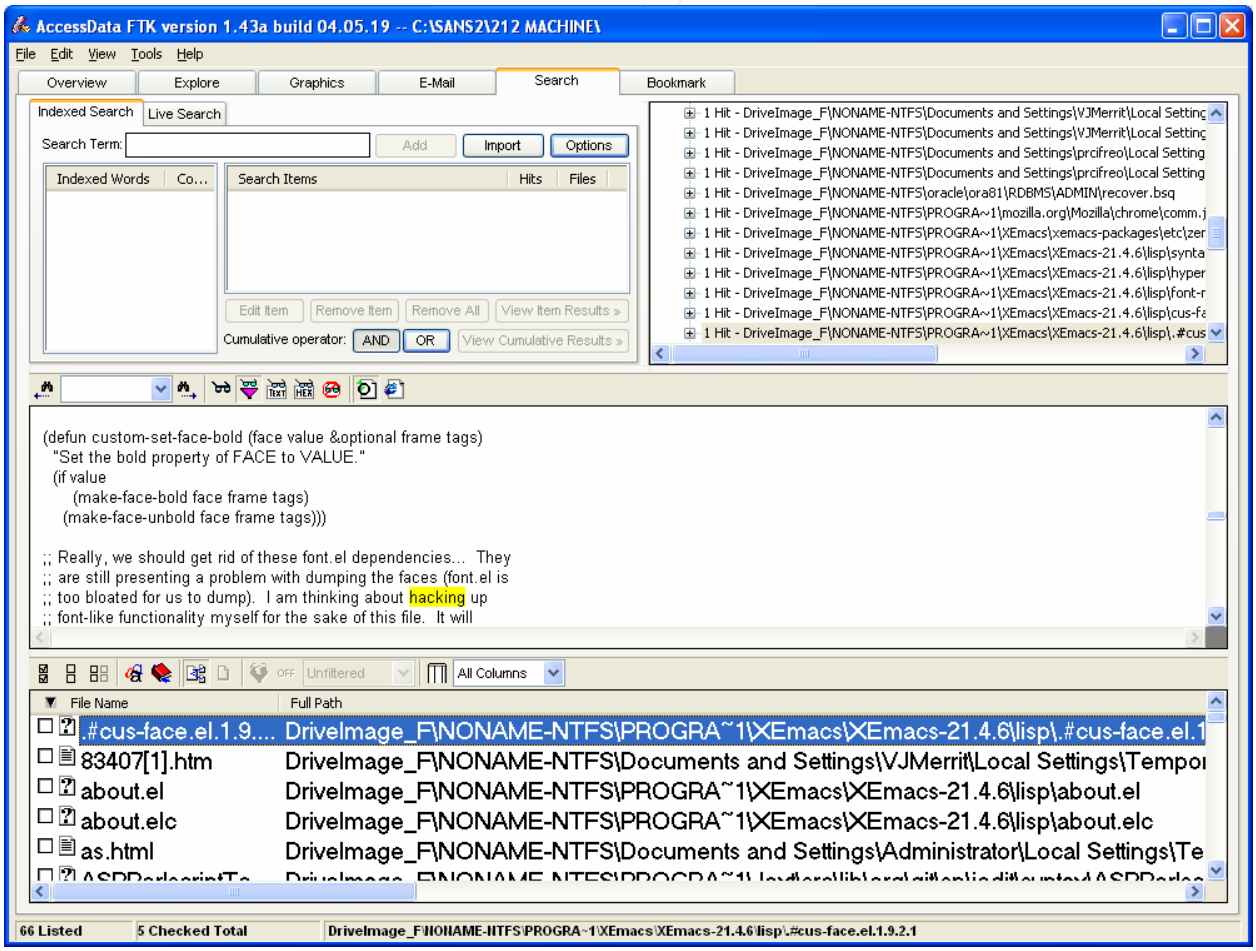
All carved TIF/PNG were random pictures from the Internet (animals, nature scenes, etc.) and were not significant.

The 410 carved Word/Excel documents contained fragments of a speech by JFK, lab assignments, several resumes, and a press release by the Clinton Whitehouse dated Jan. 7, 2000 on cybersecurity.

String Search

Several string searches were conducted against the drive image, using FTK. String searches in FTK take two forms—indexed searches and "live" searches. Indexed searches provide virtually instantaneous results, since FTK built a database of the locations of all the words in the index during initial preprocessing of the target drive. "Live" searching examines the entire drive image in real time, which can be very time-consuming. Live searches must be used for strings that are not in the index and for regular expression-based searches, however.

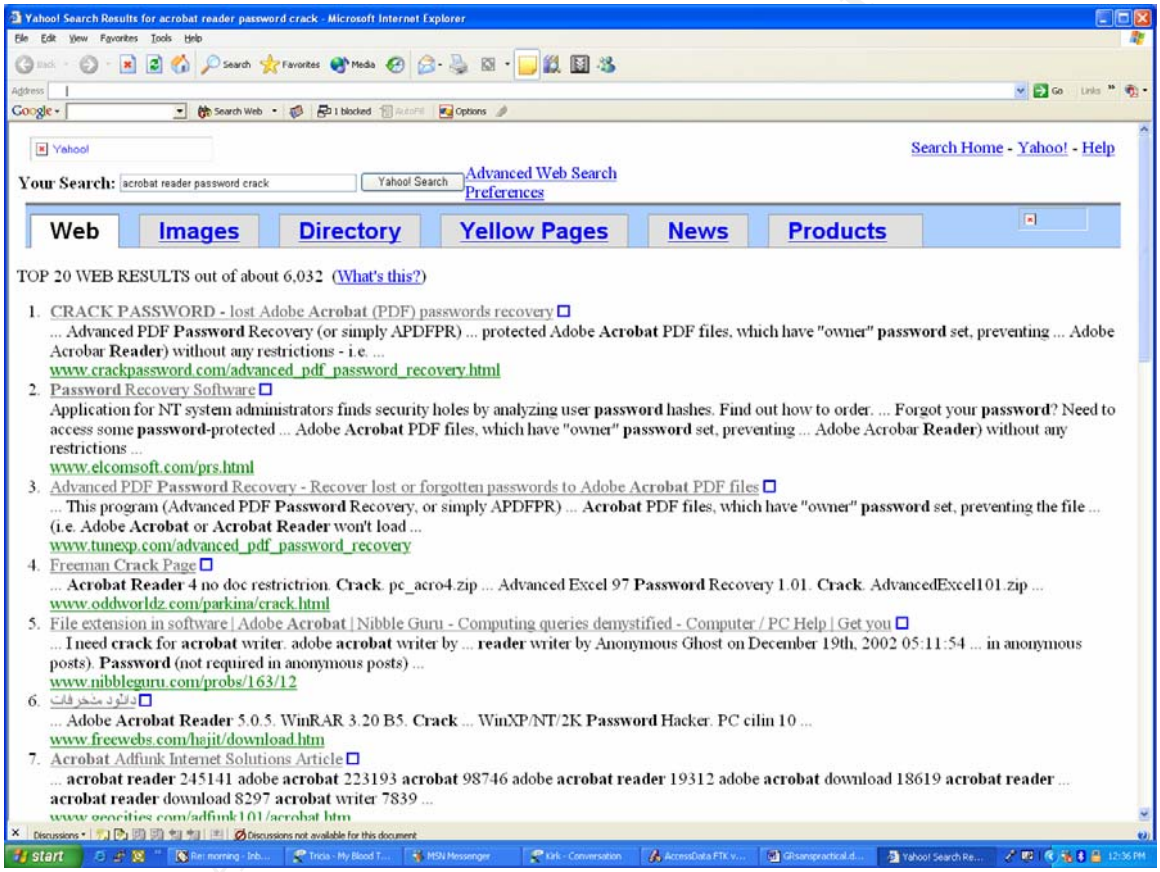
Searches in FTK are carried out using the Search dialog, accessible with a single click from the main FTK screen. The search dialog looks like the following. Search terms can be combined using AND and OR operations. Hits are displayed in the right upper corner, and clicking on a hit displays the match (in context) in the center of the screen:



A string search for “hacking” yielded no useful results—most hits were on XEMACS Lisp files. This keyword was used to see if any documents related to hacking were present on the target.

A string search on “cracking” yielded a hit in free space on a web page fragment describing the FDA “cracking down” on dietary supplements. This keyword was used to see if any documents related to cracking copyrighted software were present on the target.

A search on “cracked” + “download” yielded one hit in XXXerrit’s IE browser cache—a search for a crack for Adobe Reader password protection. This is not particularly alarming. The matching web page is shown below:



String searches did yield some additional evidence of use of the laboratory computers for viewing pornographic images. A search on keyword “porn” yielded 246 hits, most of which were fragments of web pages that looked like popups. One of the hits was on a cached web page that included some of the images by user XXMOSES, responsible for the two extensive sets of pornographic images described earlier. The following HTML fragment was recovered, with creation date 11:21:53am on 10/24/2003, tying it to the images beginning with “banner463.jpg” in the first surfing session, described earlier. Searches along these lines were not continued, because there is now a strong reason to believe that the primary offender is XXMOSES.

Welcome to bangthumbs.com

Brought to you by the makers of the World Famous Bang Bus

Amateurs Updated: Oct 24 2,120 links of hometown girls getting dirty	**Asian Girls** Updated: Oct 24 426 links of pretty asian
Big Tits Updated: Apr 7 1,548 links of beautiful, bouncing, bountiful tits	**Chubby Girls** Updated: Oct 24 163 links of Fat Fantasies
Ebony Updated: Oct 24 416 links of hard black wood	**Hardcore** Updated: Oct 24 3,398 links of down and dirty fucking
Hot Girls Updated: Oct 24 1,263 links of Maxim's Hometown Hotties	**Latinas** Updated: Oct 24 325 links of Meet Our Naughty Latinas
Lesbians Updated: Oct 24 1,460 links of girls who admit they like it	**Masturbation** Updated: Oct 24 581 links of bitches making the yellow pages proud
Mature Milfs Updated: Oct 24 702 links of somebody's mom getting fucked	**She Males** Updated: Oct 24 299 links of chicks with dicks and guys with tits
Softcore Updated: Oct 24 492 links of An erotic guide to quality softcore ...	**Teen** Updated: Oct 24 5,743 links of bitches who just got their papers

Referring Sites

1	Jpeg 4 Free	36211
2	Hammer Video	35231
3	Frog Sex	9250
4	MadThumbs	7226
5	Clip Directory	6734
6	Porn View	6665
7	88by88	5911
8	Cow List	5701
9	Pic Hunter	5584
10	teeniesxxx	4043

Trade Quality Traffic here

More categories here in the Archive - 75,000+ Galleries & Links

Search

CHAT BOARD MEET SOMEBODY NOW!

Now you have a place to chat! join other surfers, make friends and have fun!

Oct 24	Oct 24	Oct 24	Oct 24	Oct 24	Oct 24
Oct 24	Oct 24	Oct 24	Oct 24	Oct 24	Oct 24
Oct 24	Oct 24	Oct 24	Oct 24	Oct 24	Oct 24
Oct 24	Oct 24	Oct 24	Oct 24	Oct 24	Oct 24
Oct 24	Oct 24	Oct 24	Oct 24	Oct 24	Oct 24
Oct 24	Oct 24	Oct 24	Oct 24	Oct 24	Oct 24
Oct 24	Oct 24	Oct 24	Oct 24	Oct 24	Oct 24
Oct 24	Oct 24	Oct 24	Oct 24	Oct 24	Oct 24

Referring Sites

sp	Dirty Zone	3607
11	XNXX	3226
12	Day Porno	2529
13	babes2free	1640
14	Amateur Curves	1634
15	Teenage Bus	1598
16	pussyslotcom	1537
17	All Sexy Pics	1485
18	Cool Teens	1324
19	Teen Peak	1298
20	thumberland	1292
21	Teeniemovies	1206
22	BunnyTeens	1148
23	The XXX Archive	1148
24	bikini-thong-lingerie	1148
25	Worldsexlist	1122
26	SexMaxx	1115
27	Movie Shark	1010
28	SearchGals	1003
29	TeenieJunky	912
30	100 Free Sex Movies	877
31	FREE TEEN SEX PICS	806
32	JudsMovies	764
33	Teenie Files	713
34	Red Hot Honeys	701
35	dirtysexthumbs	659
36	Thumbs XXX	645
37	hollandlove	641
38	Skunklist dot Com	630
39	SperMatrix	607
40	elephantlist	584

Powered by UCJ Version IV

My personal bookmarks

While this is clearly the best site on the web ;-), I still like to check out porn at a few other select websites. Here I list the sites that I personally recommend so once you are done here, go to one of these and bust another nut!

Hammer Video	**TEENIE JUNKY**	**Teenie**	**XNXX**

The movie post site I have found.	Great Free teens pictures, and movies.	**Movies** Free Big Movies	Multi Categories Quality TGP

Oct 23	Oct 23	Oct 23	Oct 23	Oct 23	Oct 23
Oct 23	Oct 23	Oct 23	Oct 23	Oct 23	Oct 23
Oct 23	Oct 23	Oct 23	Oct 23	Oct 23	Oct 23
Oct 23	Oct 23	Oct 23	Oct 23	Oct 23	Oct 23
Oct 23	Oct 23	Oct 23	Oct 23	Oct 23	Oct 23
Oct 23	Oct 23	Oct 23	Oct 23	Oct 23	Oct 23
Oct 23	Oct 23	Oct 23	Oct 23	Oct 23	Oct 23
Oct 23	Oct 23	Oct 23	Oct 23	Oct 23	Oct 23

Referring Sites

41	Mature XXX Sluts	536
42	puppykibble	499
43	40thumbs	485
44	hot girls	389
45	Jamies Galleries	340
46	Blue Thumbs	307
47	Free XXX Zone	258
48	Update Daily	230
49	Panty Stuffing	223
50	Free Sex Portal	213
51	Teens Vids	196
52	xxbeauties	192
53	marvies	138
54	waysex	125
55	Sexxxthumb	123
56	Perfect teenies	104
57	Hot Sex Orgy	102
58	WorldSex	101
59	Sexusworld	5
60		
61		
62		
63		
64		
65		
66		
67		
68		
69		
70		

Powered by UCJ Version IV

Referring Sites

71

72

Oct 22	Oct 22	Oct 22	Oct 22	Oct 22	Oct 22
Oct 22	Oct 22	Oct 22	Oct 22	Oct 22	Oct 22
Oct 22	Oct 22	Oct 22	Oct 22	Oct 22	Oct 22
Oct 22	Oct 22	Oct 22	Oct 22	Oct 22	Oct 22
Oct 22	Oct 22	Oct 22	Oct 22	Oct 22	Oct 22
Oct 22	Oct 22	Oct 22	Oct 22	Oct 22	Oct 22
Oct 22	Oct 22	Oct 22	Oct 22	Oct 22	Oct 22
Oct 22	Oct 22	Oct 22	Oct 22	Oct 22	Oct 22

73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

Powered by UCJ Version IV

Next page of updates >>

[Bangthumbs Archive - over 80,000+ pretty girls]

My personal bookmarks			
While this is clearly the best site on the web ;-), I still like to check out porn at a few other select websites. Here I list the sites that I personally recommend so once you are done here, go to one of these and bust another nut!			
MMM100 great movies, huge archive of links.	**Link Blender** More Free Quality galleries	**The Elephant List** Free pics, movies and huge archive of links.	**World Sex** Natural tits, Round Asses and Cute Girls

Conclusions

Analysis of the machine revealed that, in general, the flavor of the policy document and system banners were adhered to by laboratory users using this particular machine.

Applications were installed on the machine under investigation over a few fixed timeframes. There is no evidence that unauthorized copyrighted software was installed.

The vast majority of the content on the machine was for class assignments. No collections of MP3s or digital media were noted. No online chatting software was detected. Keyword searches and a casual scan of HTML documents in the user's browser caches revealed nothing to indicate that unauthorized software or other copyrighted material such as MP3s were downloaded using the machine, supporting the theory that overall, use of the machine falls squarely within the acceptable use guidelines.

Very few files were marked deleted and very few files were present in the users' recycle bins. This supports a view that the machine was typically used in a "straightforward" manner, with users making little effort to conceal their activities.

No compromises of the machine were evident. FTK's Known File Filter (KFF) failed to detect any known malware. Further, a full system scan by Norton Antivirus 2004 with the latest version of the Symantec virus definitions on a copy of the drive (installed in a test machine) failed to detect any viruses, worms, or other malware.

On the other hand, a single user ("XXXOSES") clearly used the machine to view hardcore pornographic materials on two occasions, as documented in the Media Analysis section. Extensive file carving to discover other inappropriate material failed to uncover any images or other digital media of the sort viewed by XXXOSES, though a few mildly inappropriate images were recovered. These carved images may be associated with the activities of XXXOSES. The failure of file carving to reveal much interesting content (combined with little evidence that a secure file wiping utility was ever installed) supports the theory that the machine was not typically used for viewing pornographic materials.

References

[1] National Drug Intelligence Center, http://www.usdoj.gov/ndic/about.htm.

[2] National Software Reference Library, http://www.nsrl.nist.gov/.

[3] Force 2.0.8, http://downloads.zdnet.co.uk/0,39025604,39066423s,00.htm.

Part 3: Legal Issues of Incident Handling

Assumption: John Price (from Part 1) was distributing copyrighted material on publicly available systems.

A. Based on the type of material John Price was distributing, what, if any, laws have been broken based upon the distribution?

The Copyright Act of 1976 gives the copyright holder exclusive rights to create and distribute copies of musical recordings (though typically some or all of these rights are transferred to a record company). Distributing copyrighted material violates the Copyright Act of 1976 as well as the newer Digital Millennium Copyright Act (DMCA). Note that even if John Price owns the CDs from which the MP3s were ripped, his distribution of the MP3 is illegal. The First Sale doctrine of the Copyright Act allows the sale or distribution (that is, a transfer of ownership) or destruction of a legal copy of a recording, but the transfer of ownership and possession are closely related. In simpler terms, a recording that one owns legally may be sold or given away once, but the right to distribute disappears once the first copy is out the door.

B. What would the appropriate steps be if you discovered this information on your systems? Cite specific statutes.

John Price is an employee of the company performing the investigation. Assuming that the computer account provided to John is a result of his special relationship with the company (that is, the company would not provide an account of the same type to an arbitrary member of the general public), then the Electronic Communications Privacy Act does not apply and all materials from the investigation can be turned over the law enforcement. This information is taken directly from the GIAC course materials.

C. In the event your corporate counsel decides not to pursue the matter any further at this point, what steps should you take to ensure that any evidence you collect can be admissible in proceedings should the situation change?

The original floppy and drive image must be preserved, along with the investigative notes. A "chain of custody" for these evidentiary items, even in a corporate setting, is best preserved.

D. How would your actions change if your investigation disclosed that John Price was distributing child pornography?

In this event, the entire investigation and any evidence collected would be turned over immediately to law enforcement. Child pornography is contraband in the United States and its possession by citizens is strictly prohibited, thus any further corporate investigation would be inappropriate. The Federal Obscenity Statute (18 U.S.C. § 1462)

and Federal Child Pornography Statute: (18 U.S.C. § 2252) are often used in the prosecution of child pornography.