Waterford Institute of Technology

System Forensics 2

File System Investigation

Jack Donoghue

BSc (Hons) in Computer Forensics and Security – Year 2

20072172

Donoghuej97@gmail.com

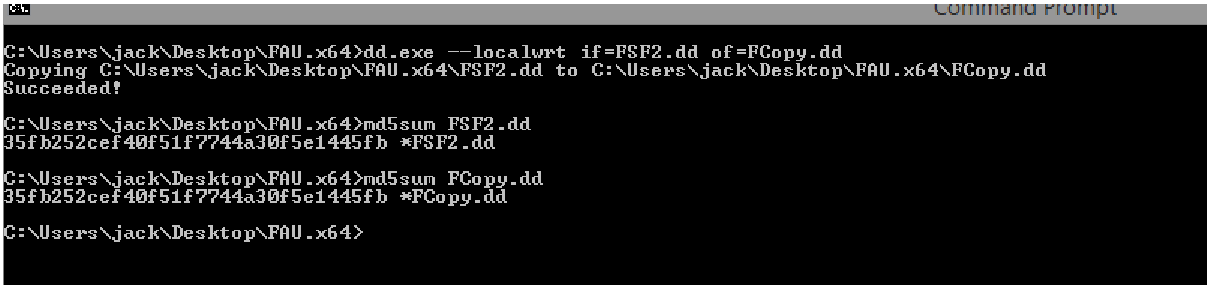
Contents

Introduction

This report will focus on key evidence recovered from the USB drive of former employee Anne O' Brien, the process of recovery in relation to evidence will be strictly outlined in the coming pages and will seek to inform the reader of the tools used to recover and the methods carried out, the original data from the USB drive will not be altered or tampered with throughout the process of this investigation and all evidence recovered will be from a controlled copy of the original disk image.

Forensic Duplicate

As mentioned above I first created a safe forensic duplicate from which I would carry out my investigation on, this is done to reduce risk of contamination of the file system and therefore the evidence contained. I began this process by using Fau.x64 and the dd.exe localwrt command to copy the original file to workable file. I then ensured both files were a complete replica by checking the md5sum of both files, the results can be seen below, both files returned the exact same md5sums proving that they were sound replicas.

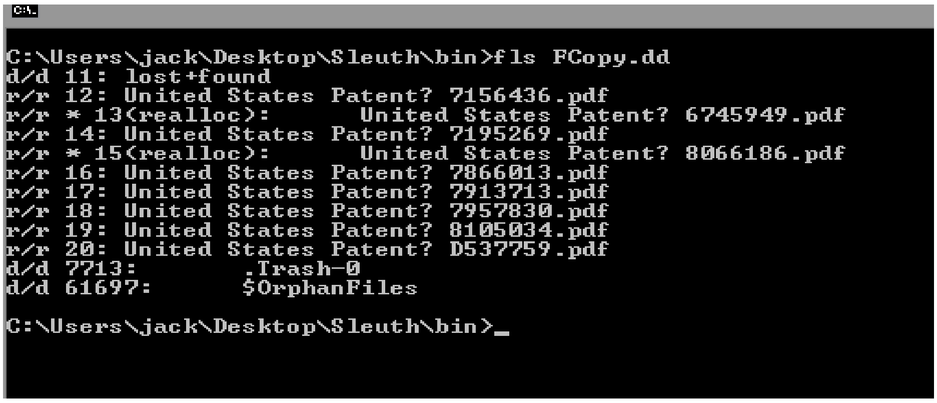


```
C:\Users\jack\Desktop\FAU.x64>dd.exe --localwrt if=FSF2.dd of=FCopy.dd
Copying C:\Users\jack\Desktop\FAU.x64\FSF2.dd to C:\Users\jack\Desktop\FAU.x64\FCopy.dd
Succeeded!

C:\Users\jack\Desktop\FAU.x64>md5sum FSF2.dd
35fb252cef40f51f7744a30f5e1445fb *FSF2.dd

C:\Users\jack\Desktop\FAU.x64>md5sum FCopy.dd
35fb252cef40f51f7744a30f5e1445fb *FCopy.dd

C:\Users\jack\Desktop\FAU.x64>
```

Figure 1: Forensic copy and md5sums

Initial Inspection

After I was satisfied with the trustworthiness of the forensic duplicate I proceeded to look inside of the copied file, upon first inspection of the file I was presented with three separate directories and a total of nine PDF files relating to United States Patents, I found this information by using the fls command (Sleuth) as seen below.

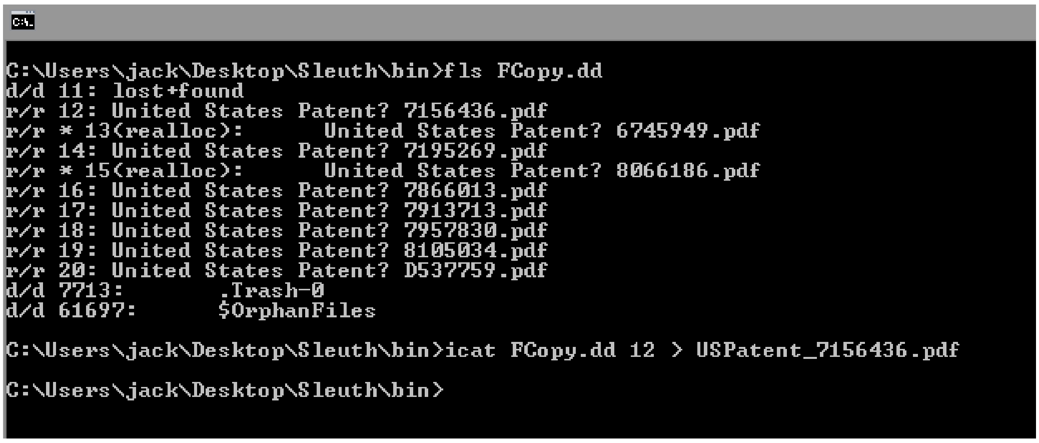


```
C:\Users\jack\Desktop\Sleuth\bin>fls FCopy.dd
d/d 11: lost+found
r/r 12: United States Patent? 7156436.pdf
r/r * 13(realloc):      United States Patent? 6745949.pdf
r/r 14: United States Patent? 7195269.pdf
r/r * 15(realloc):      United States Patent? 8066186.pdf
r/r 16: United States Patent? 7866013.pdf
r/r 17: United States Patent? 7913713.pdf
r/r 18: United States Patent? 7957830.pdf
r/r 19: United States Patent? 8105034.pdf
r/r 20: United States Patent? D537759.pdf
d/d 7713:       .Trash-0
d/d 61697:      $OrphanFiles

C:\Users\jack\Desktop\Sleuth\bin>_
```

Figure 2: Initial Inspection

Evidence Retrieval Process

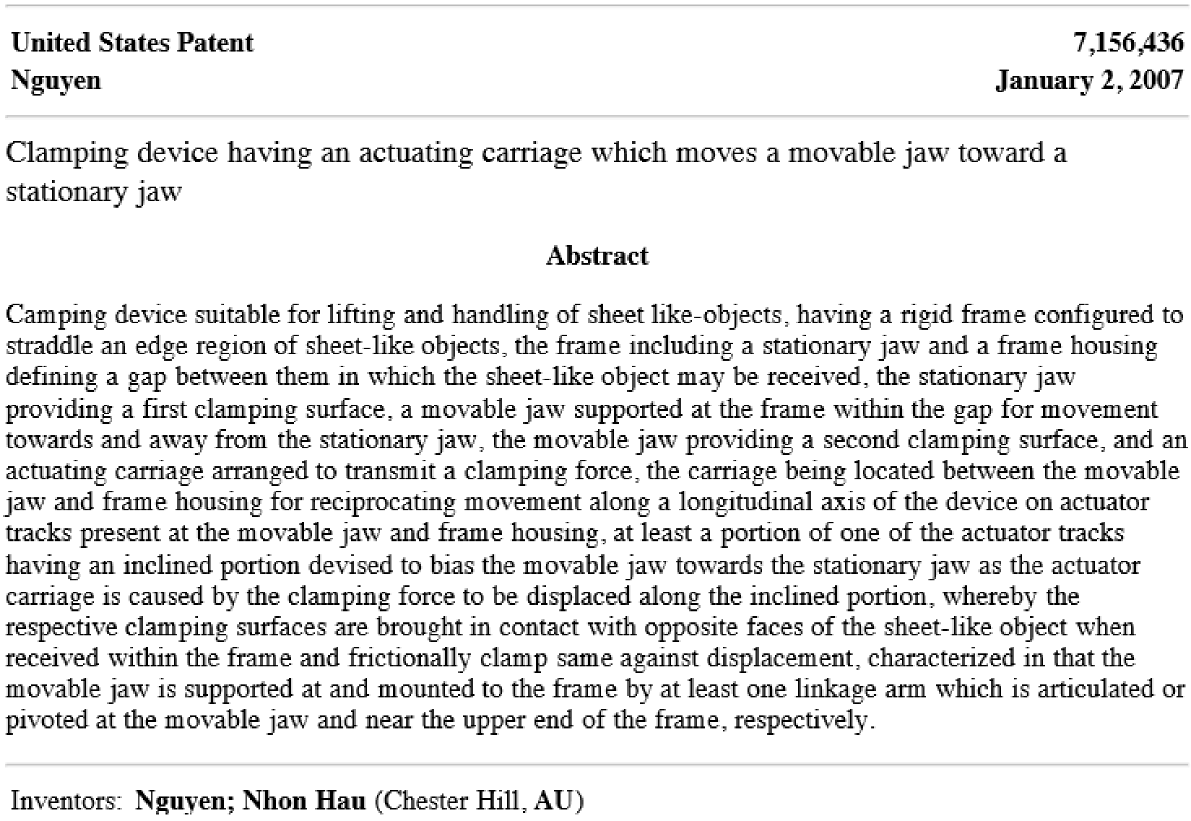
The evidence found in the main directory of the USB drive contained multiple PDFs relating to United States Patents, I ported the PDFs out of the forensic duplicate and onto my PC by using the icat command (Sleuth), Image below displays the process used to port the contents of the PDF to a readable format on my PC.



```
C:\Users\jack\Desktop\Sleuth\bin>fls FCopy.dd
d/d 11: lost+found
r/r 12: United States Patent? 7156436.pdf
r/r * 13(realloc):      United States Patent? 6745949.pdf
r/r 14: United States Patent? 7195269.pdf
r/r * 15(realloc):      United States Patent? 8066186.pdf
r/r 16: United States Patent? 7866013.pdf
r/r 17: United States Patent? 7913713.pdf
r/r 18: United States Patent? 7957830.pdf
r/r 19: United States Patent? 8105034.pdf
r/r 20: United States Patent? D537759.pdf
d/d 7713:       .Trash-0
d/d 61697:      $OrphanFiles

C:\Users\jack\Desktop\Sleuth\bin>icat FCopy.dd 12 > USPatent_7156436.pdf

C:\Users\jack\Desktop\Sleuth\bin>
```

Figure 3: Evidence on USB drive

After this was done I was able to retrieve the contents of the file by entering the Bin in the Sleuth folder.



United States Patent	**7,156,436**
Nguyen	**January 2, 2007**

Clamping device having an actuating carriage which moves a movable jaw toward a stationary jaw

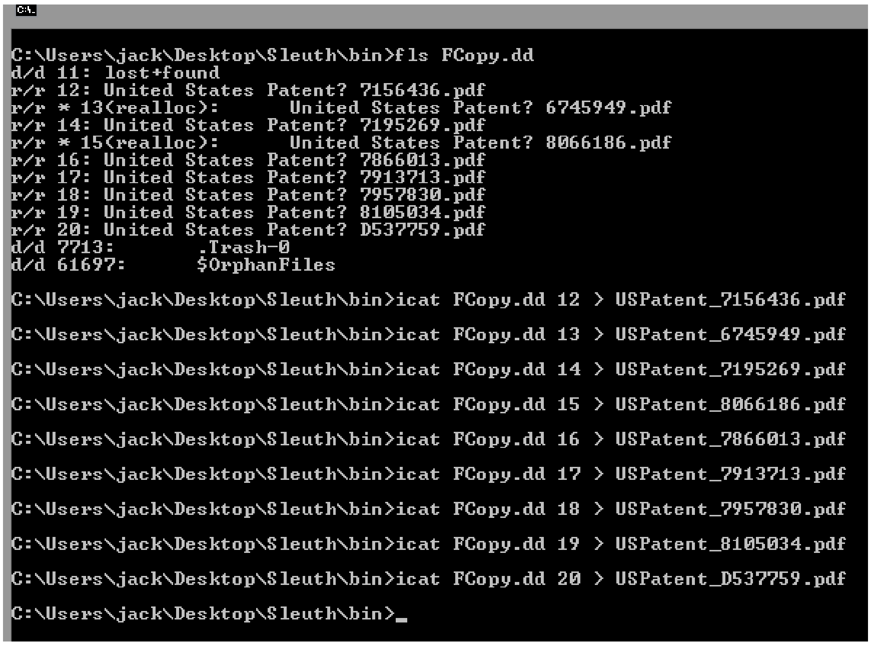
Abstract

Camping device suitable for lifting and handling of sheet like-objects, having a rigid frame configured to straddle an edge region of sheet-like objects, the frame including a stationary jaw and a frame housing defining a gap between them in which the sheet-like object may be received, the stationary jaw providing a first clamping surface, a movable jaw supported at the frame within the gap for movement towards and away from the stationary jaw, the movable jaw providing a second clamping surface, and an actuating carriage arranged to transmit a clamping force, the carriage being located between the movable jaw and frame housing for reciprocating movement along a longitudinal axis of the device on actuator tracks present at the movable jaw and frame housing, at least a portion of one of the actuator tracks having an inclined portion devised to bias the movable jaw towards the stationary jaw as the actuator carriage is caused by the clamping force to be displaced along the inclined portion, whereby the respective clamping surfaces are brought in contact with opposite faces of the sheet-like object when received within the frame and frictionally clamp same against displacement, characterized in that the movable jaw is supported at and mounted to the frame by at least one linkage arm which is articulated or pivoted at the movable jaw and near the upper end of the frame, respectively.

Inventors: **Nguyen; Nhon Hau** (Chester Hill, **AU**)

Figure 4: US Patent 7156436 PDF

I then retrieved the rest of the PDFs as seen below.



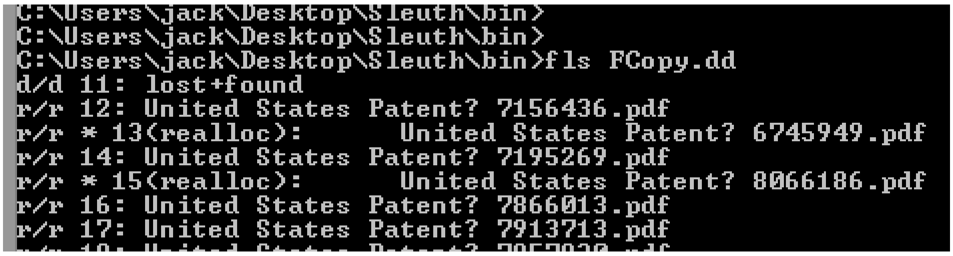
```
C:\Users\jack\Desktop\Sleuth\bin>fls FCopy.dd
d/d 11: lost+found
r/r 12: United States Patent? 7156436.pdf
r/r * 13(realloc):      United States Patent? 6745949.pdf
r/r 14: United States Patent? 7195269.pdf
r/r * 15(realloc):      United States Patent? 8066186.pdf
r/r 16: United States Patent? 7866013.pdf
r/r 17: United States Patent? 7913713.pdf
r/r 18: United States Patent? 7957830.pdf
r/r 19: United States Patent? 8105034.pdf
r/r 20: United States Patent? D537759.pdf
d/d 7713:       .Trash-0
d/d 61697:      $OrphanFiles

C:\Users\jack\Desktop\Sleuth\bin>icat FCopy.dd 12 > USPatent_7156436.pdf

C:\Users\jack\Desktop\Sleuth\bin>icat FCopy.dd 13 > USPatent_6745949.pdf

C:\Users\jack\Desktop\Sleuth\bin>icat FCopy.dd 14 > USPatent_7195269.pdf

C:\Users\jack\Desktop\Sleuth\bin>icat FCopy.dd 15 > USPatent_8066186.pdf

C:\Users\jack\Desktop\Sleuth\bin>icat FCopy.dd 16 > USPatent_7866013.pdf

C:\Users\jack\Desktop\Sleuth\bin>icat FCopy.dd 17 > USPatent_7913713.pdf

C:\Users\jack\Desktop\Sleuth\bin>icat FCopy.dd 18 > USPatent_7957830.pdf

C:\Users\jack\Desktop\Sleuth\bin>icat FCopy.dd 19 > USPatent_8105034.pdf

C:\Users\jack\Desktop\Sleuth\bin>icat FCopy.dd 20 > USPatent_D537759.pdf

C:\Users\jack\Desktop\Sleuth\bin>_
```

Figure 5: Extraction of all Patents

Each file was extracted by first selecting the forensic copy and then selecting the inode number, I then name the file and assign it a file signature.

Points of Interest on the USB Drive

There were two patents in pdf format that were removed from the main directory of the USB, as seen below these PDFs are missing from their original directory.

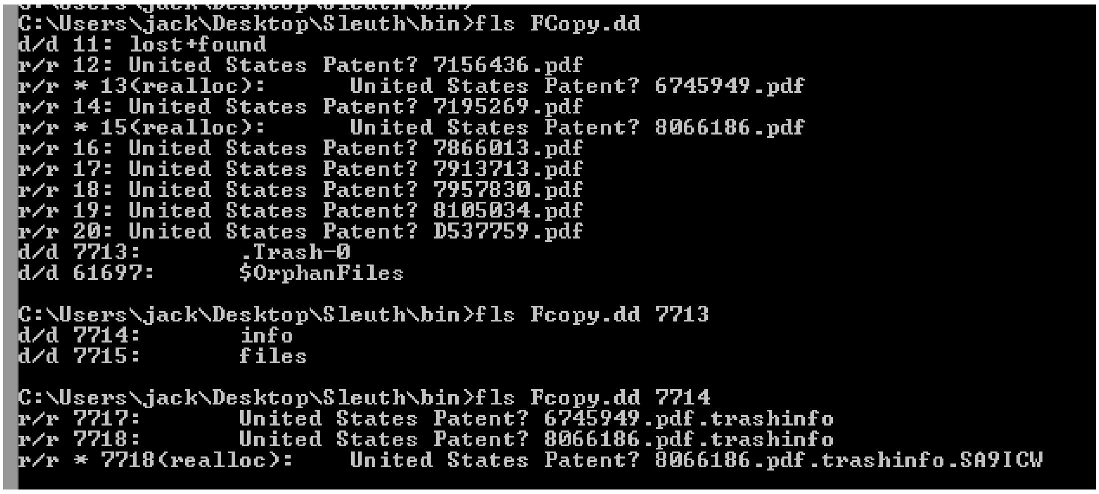


```
C:\Users\jack\Desktop\Sleuth\bin>
C:\Users\jack\Desktop\Sleuth\bin>
C:\Users\jack\Desktop\Sleuth\bin>fls FCopy.dd
d/d 11: lost+found
r/r 12: United States Patent? 7156436.pdf
r/r * 13(realloc):      United States Patent? 6745949.pdf
r/r 14: United States Patent? 7195269.pdf
r/r * 15(realloc):      United States Patent? 8066186.pdf
r/r 16: United States Patent? 7866013.pdf
r/r 17: United States Patent? 7913713.pdf
r/r 18: United States Patent? 7957830.pdf
```

Figure 6 File Removal

The asterisk followed by the inode number and a (realloc) notification show that the files have been removes or places elsewhere in another directory or location.

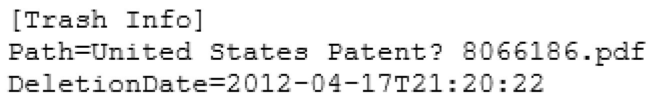
Upon further inspection of the USB drive the two PDFs in question were located in the .Trash-0 folder of the file system, this indicates an attempt by Anne to perhaps dispose of potentially incriminating documents that she had in her possession.



```
C:\Users\jack\Desktop\Sleuth\bin>
C:\Users\jack\Desktop\Sleuth\bin>fls FCopy.dd
d/d 11: lost+found
r/r 12: United States Patent? 7156436.pdf
r/r * 13(realloc):      United States Patent? 6745949.pdf
r/r 14: United States Patent? 7195269.pdf
r/r * 15(realloc):      United States Patent? 8066186.pdf
r/r 16: United States Patent? 7866013.pdf
r/r 17: United States Patent? 7913713.pdf
r/r 18: United States Patent? 7957830.pdf
r/r 19: United States Patent? 8105034.pdf
r/r 20: United States Patent? D537759.pdf
d/d 7713:       .Trash-0
d/d 61697:      $OrphanFiles

C:\Users\jack\Desktop\Sleuth\bin>fls Fcopy.dd 7713
d/d 7714:       info
d/d 7715:       files

C:\Users\jack\Desktop\Sleuth\bin>fls Fcopy.dd 7714
r/r 7717:       United States Patent? 6745949.pdf.trashinfo
r/r 7718:       United States Patent? 8066186.pdf.trashinfo
r/r * 7718(realloc):    United States Patent? 8066186.pdf.trashinfo.SA9ICW
```

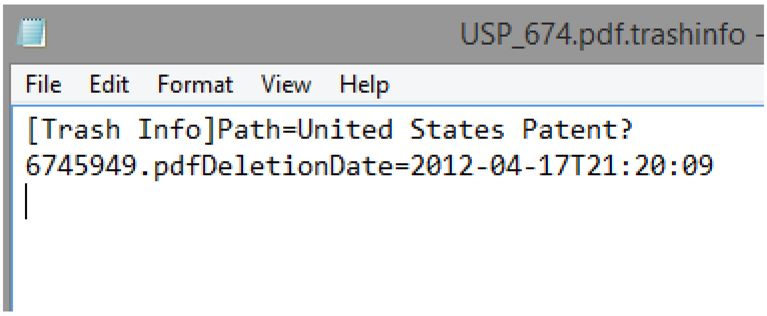
Figure 7 Recovery of the deleted metadata

The two files were still able to be recovered along with all of their contents, however by accessing the .trashinfo files left behind by the two files that were removed I was able to obtain valuable evidence as to the times of deletion of both files.



```
[Trash Info]
Path=United States Patent? 8066186.pdf
DeletionDate=2012-04-17T21:20:22
```

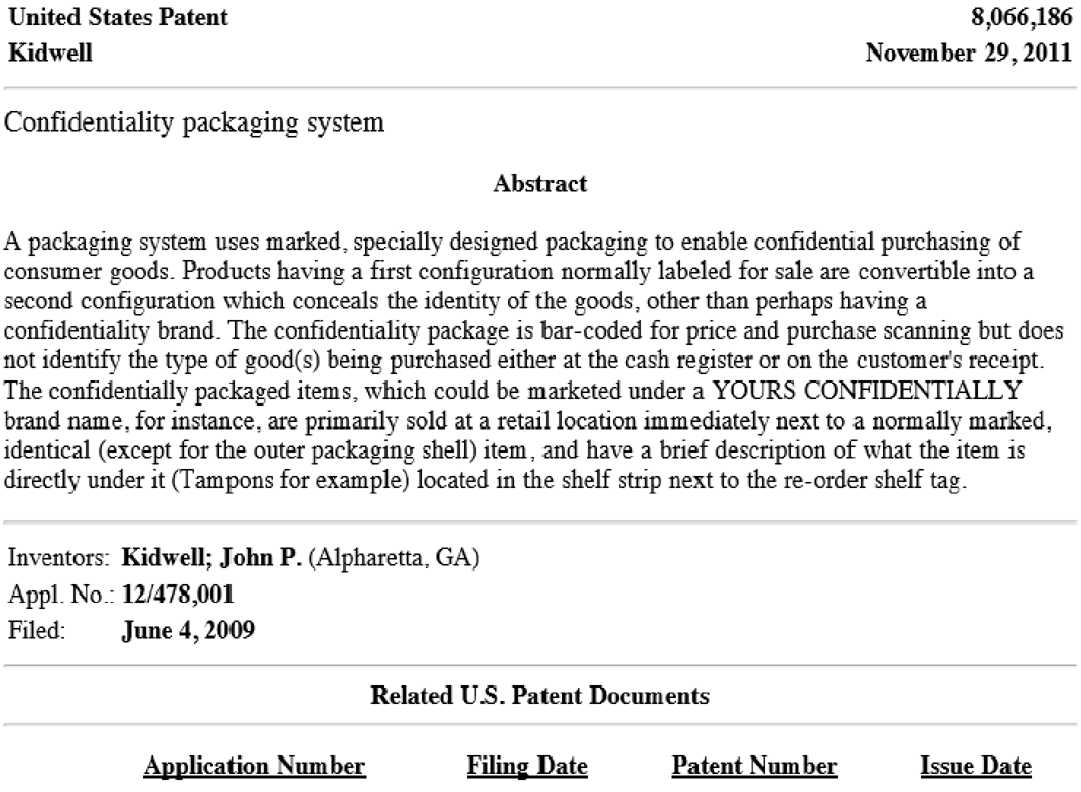
Figure 8 Recovered Info (Deletion 2)



```
[Trash Info]Path=United States Patent?
6745949.pdfDeletionDate=2012-04-17T21:20:09
```

Figure 1Figure 9 Recovered Info (Deletion 2)

This information allows us to see that as well as being in possession of multiple PDFs containing patented information, there was also an attempt to delete these two PDF's as well as their content, however as the files were improperly deleted I was also able to recover both of the contents of the PDFs as seen in the next page.



United States Patent **8,066,186**
Kidwell **November 29, 2011**

Confidentiality packaging system

Abstract

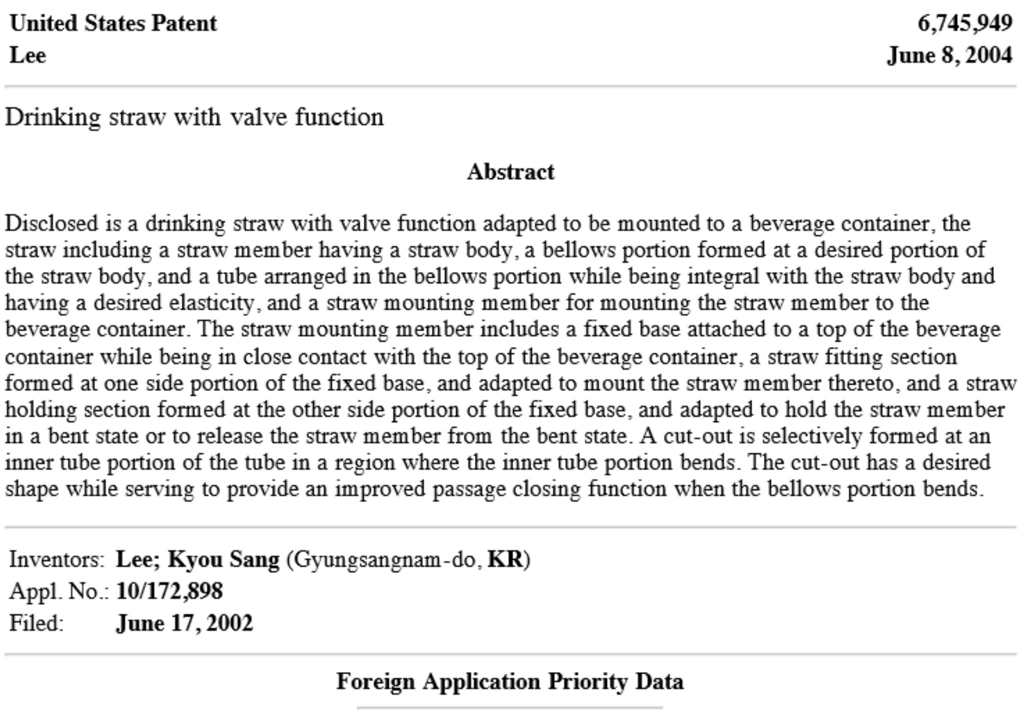
A packaging system uses marked, specially designed packaging to enable confidential purchasing of consumer goods. Products having a first configuration normally labeled for sale are convertible into a second configuration which conceals the identity of the goods, other than perhaps having a confidentiality brand. The confidentiality package is bar-coded for price and purchase scanning but does not identify the type of good(s) being purchased either at the cash register or on the customer's receipt. The confidentially packaged items, which could be marketed under a YOURS CONFIDENTIALLY brand name, for instance, are primarily sold at a retail location immediately next to a normally marked, identical (except for the outer packaging shell) item, and have a brief description of what the item is directly under it (Tampons for example) located in the shelf strip next to the re-order shelf tag.

Inventors: **Kidwell; John P.** (Alpharetta, GA)
Appl. No.: **12/478,001**
Filed: **June 4, 2009**

Related U.S. Patent Documents

Application Number	Filing Date	Patent Number	Issue Date

Figure 9 Retrieved PDF (1)



United States Patent **6,745,949**
Lee **June 8, 2004**

Drinking straw with valve function

Abstract

Disclosed is a drinking straw with valve function adapted to be mounted to a beverage container, the straw including a straw member having a straw body, a bellows portion formed at a desired portion of the straw body, and a tube arranged in the bellows portion while being integral with the straw body and having a desired elasticity, and a straw mounting member for mounting the straw member to the beverage container. The straw mounting member includes a fixed base attached to a top of the beverage container while being in close contact with the top of the beverage container, a straw fitting section formed at one side portion of the fixed base, and adapted to mount the straw member thereto, and a straw holding section formed at the other side portion of the fixed base, and adapted to hold the straw member in a bent state or to release the straw member from the bent state. A cut-out is selectively formed at an inner tube portion of the tube in a region where the inner tube portion bends. The cut-out has a desired shape while serving to provide an improved passage closing function when the bellows portion bends.

Inventors: **Lee; Kyou Sang** (Gyungsangnam-do, **KR**)
Appl. No.: **10/172,898**
Filed: **June 17, 2002**

Foreign Application Priority Data

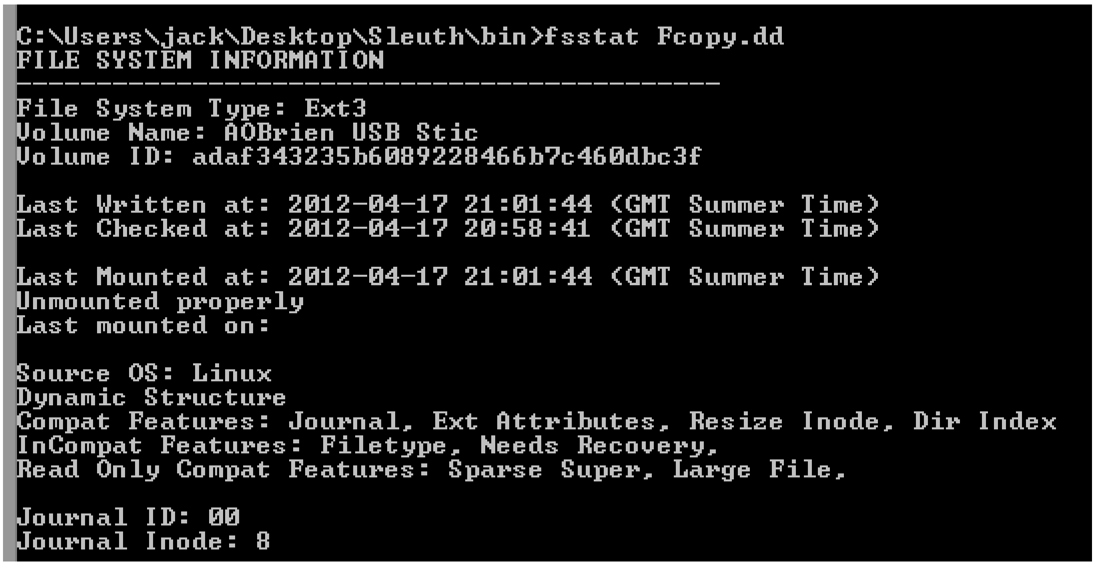
Figure 10 Retrieved PDF (2)

Analysis of the File System

The file system used on Anne's USB drive is ext3[1], this older Linux based file system is commonly found on USB and other low end storage devices. This file system has a maximum storage capacity of 2 TB to 32 TB which is suitable for a memory storage medium such as a USB drive, as it is unlikely in the near future to be able to store such a large amount of data on a USB drive. Benefits of using an Ext3 file system for a memory storage medium are plenty, one of the main improvements from the Ext2 model was the introduction of Journaling which has allowed for data and content to be safely stored and logged in the event of a crash or in the USB's case and unexpected ejection from an end – user device such as laptop or desktop.

Looking into the USB's File System

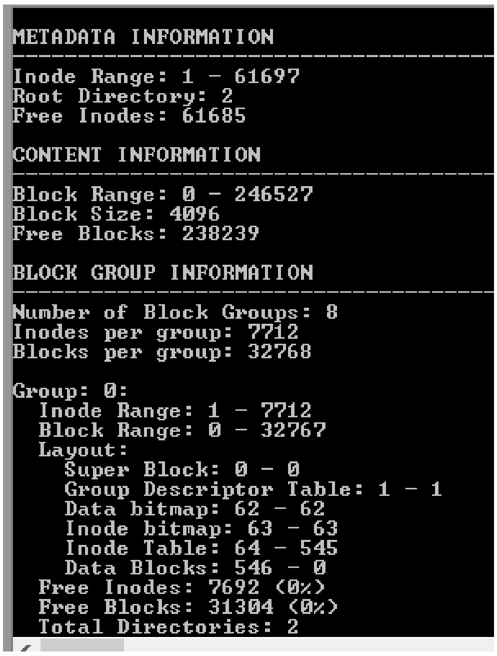
Here is an extract from an analysis of the entire file system, retrieved using the fsstat command, we can now see the file system we are dealing with (Ext3) and the name of the user 'AOBrien USB Stic'. Also available to us is other valuable information such as the last time data was written to the filesystem, this is in the form of adding or removing files from the USB. Finally we are given information relating to the operating system in use and features of the Ext3 file system.



```
C:\Users\jack\Desktop\Sleuth\bin>fsstat Fcopy.dd
FILE SYSTEM INFORMATION
--------------------------------------------
File System Type: Ext3
Volume Name: AOBrien USB Stic
Volume ID: adaf343235b6089228466b7c460dbc3f

Last Written at: 2012-04-17 21:01:44 (GMT Summer Time)
Last Checked at: 2012-04-17 20:58:41 (GMT Summer Time)

Last Mounted at: 2012-04-17 21:01:44 (GMT Summer Time)
Unmounted properly
Last mounted on:

Source OS: Linux
Dynamic Structure
Compat Features: Journal, Ext Attributes, Resize Inode, Dir Index
InCompat Features: Filetype, Needs Recovery,
Read Only Compat Features: Sparse Super, Large File,

Journal ID: 00
Journal Inode: 8
```

Figure 11 Metadata

[1] http://www.thegeekstuff.com/2011/05/ext2-ext3-ext4
ext3 details



```
METADATA INFORMATION
--------------------------------------------
Inode Range: 1 - 61697
Root Directory: 2
Free Inodes: 61685

CONTENT INFORMATION
--------------------------------------------
Block Range: 0 - 246527
Block Size: 4096
Free Blocks: 238239

BLOCK GROUP INFORMATION
--------------------------------------------
Number of Block Groups: 8
Inodes per group: 7712
Blocks per group: 32768

Group: 0:
  Inode Range: 1 - 7712
  Block Range: 0 - 32767
  Layout:
    Super Block: 0 - 0
    Group Descriptor Table: 1 - 1
    Data bitmap: 62 - 62
    Inode bitmap: 63 - 63
    Inode Table: 64 - 545
    Data Blocks: 546 - 0
  Free Inodes: 7692 (0%)
  Free Blocks: 31304 (0%)
  Total Directories: 2
```

Figure 12 File system information

This information is also found using the fsstat command, it details information such as Inode, block and group information, I will look at detailing the exact layout of each block group as to draw a physical map of the file systems topology.

Tabling of Block Groups

Group 0

Superblock	Group Descriptor Table	Block Bitmap	Inode Bitmap	Inode Table	Data Blocks
0 - 0	1 - 1	62 - 62	63 - 63	64 - 545	546 – 31304

Group 1

Superblock	Group Descriptor Table	Data Bitmap	Inode Bitmap	Inode Table	Data Blocks
32768-32768	32769 – 32769	32830 - 32830	32831 - 32831	32832 - 3313	33314 -65531

Group 2

Superblock	Group Descriptor Table	Data Bitmap	Inode Bitmap	Inode Table	Data Blocks
Unknown	Unknown	65535 – 65536	65537 - 65537	65538 - 66019	65538 - 94203

Group 3

Superblock	Group Descriptor Table	Data Bitmap	Inode Bitmap	Inode Table	Data Blocks
98304 - 98304	98305 - 98305	98366 - 98366	98367 – 98367	98368 - 98849	98850 -131071

Group 4

Superblock	Group Descriptor Table	Data Bitmap	Inode Bitmap	Inode Table	Data Blocks

| Unknown | Unknown |131072 – 131072 |163903- 163903|163904 - 164385|164386 - 163839

Group 5

Superblock	Group Descriptor Table	Data Bitmap	Inode Bitmap	Inode Table	Data Blocks

|163840-163840|163841-163841|163902-163902|163903-163903|163904-164385|164386-1996607

Group 6

Superblock	Group Descriptor Table	Data Bitmap	Inode Bitmap	Inode Table	Data Blocks

|Unknown |Unknown |196608-196608 |196609-196609|196610-197091|197092-229376

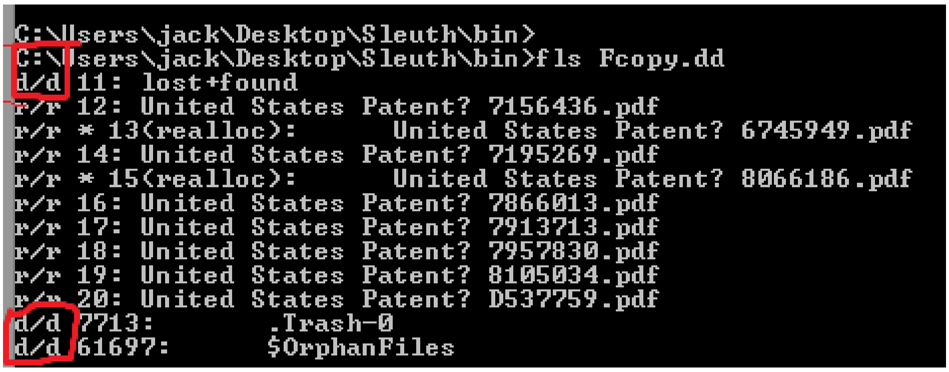
Group 7

Superblock	Group Descriptor Table	Data Bitmap	Inode Bitmap	Inode Table	Data Blocks

|229376-229376|229377-229377|229439-229439|229439-229439|229440-229921|229922-246528

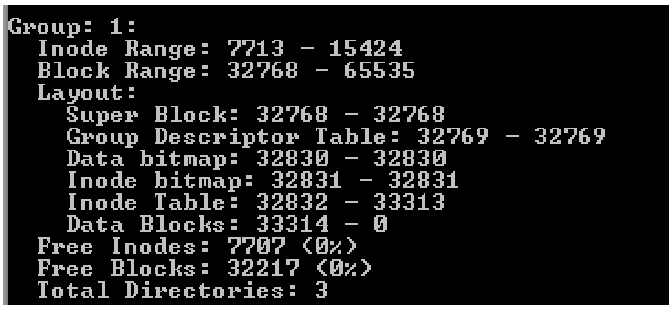
Group Contents

The contents of the groups can be seen from the metadata retrieved using the fsstat command, in Group 0 we can see that there are two directories, if we look at the initial Fcopy.dd file we see that there are three directories here, therefore Group 0 cannot be holding these three directories.



```
C:\Users\jack\Desktop\Sleuth\bin>
C:\Users\jack\Desktop\Sleuth\bin>fls Fcopy.dd
d/d 11: lost+found
r/r 12: United States Patent? 7156436.pdf
r/r * 13(realloc):      United States Patent? 6745949.pdf
r/r 14: United States Patent? 7195269.pdf
r/r * 15(realloc):      United States Patent? 8066186.pdf
r/r 16: United States Patent? 7866013.pdf
r/r 17: United States Patent? 7913713.pdf
r/r 18: United States Patent? 7957830.pdf
r/r 19: United States Patent? 8105034.pdf
r/r 20: United States Patent? D537759.pdf
d/d 7713:       .Trash-0
d/d 61697:      $OrphanFiles
```

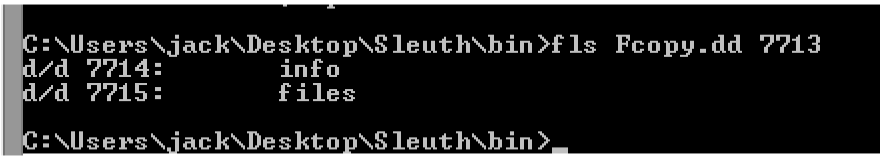
Figure 13 Three Directories (Group 1)



```
Group: 1:
  Inode Range: 7713 - 15424
  Block Range: 32768 - 65535
  Layout:
    Super Block: 32768 - 32768
    Group Descriptor Table: 32769 - 32769
    Data bitmap: 32830 - 32830
    Inode bitmap: 32831 - 32831
    Inode Table: 32832 - 33313
    Data Blocks: 33314 - 0
  Free Inodes: 7707 (0%)
  Free Blocks: 32217 (0%)
  Total Directories: 3
```

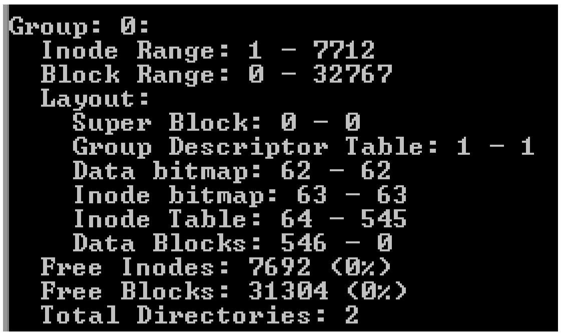
Figure 14 Shows Group 1 with 3 Directories

To find the location of Group 1 I had to search around the file system to find a memory location that had two directories being stored in it.



```
C:\Users\jack\Desktop\Sleuth\bin>fls Fcopy.dd 7713
d/d 7714:       info
d/d 7715:       files

C:\Users\jack\Desktop\Sleuth\bin>_
```

Figure 15 Two Directories (Group 0)



```
Group: 0:
  Inode Range: 1 - 7712
  Block Range: 0 - 32767
  Layout:
    Super Block: 0 - 0
    Group Descriptor Table: 1 - 1
    Data bitmap: 62 - 62
    Inode bitmap: 63 - 63
    Inode Table: 64 - 545
    Data Blocks: 546 - 0
  Free Inodes: 7692 (0%)
  Free Blocks: 31304 (0%)
  Total Directories: 2
```

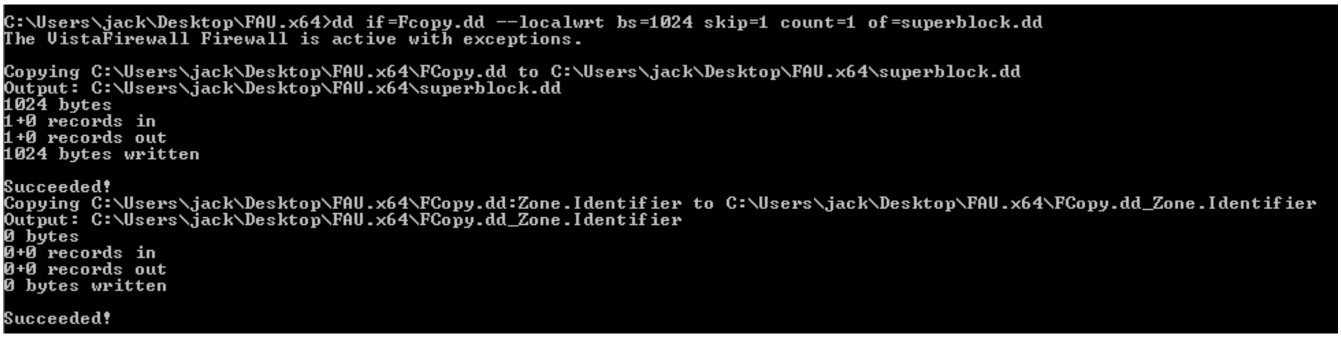
Figure 16 Group 0 with 2 Directories

Data Structures in Ext3

Superblock

The superblock in the Ext3 file system is essentially a storage device for metadata relating to the running of the entire file system, Superblocks store critical information about the file system, it also stores things such as the configuration of the file system[2]. The file systems superblock will also hold essential information for the system such as the number of blocks available for use in the system and the File System State. Because of its importance the Superblock is located at the very first block group, also known as the primary Superblock.

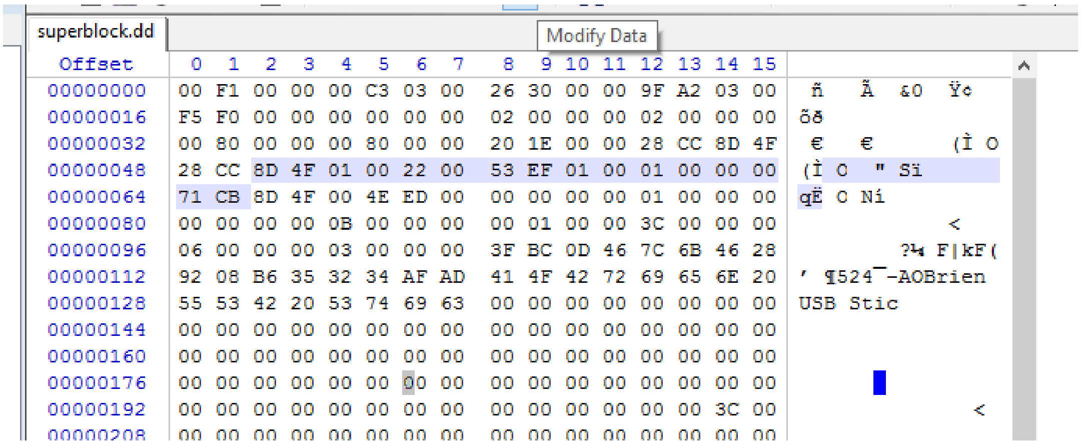
Since valuable data can be stored in the superblock I will be extracting one from the USB stick under investigation.



```
C:\Users\jack\Desktop\FAU.x64>dd if=Fcopy.dd --localwrt bs=1024 skip=1 count=1 of=superblock.dd
The VistaFirewall Firewall is active with exceptions.

Copying C:\Users\jack\Desktop\FAU.x64\FCopy.dd to C:\Users\jack\Desktop\FAU.x64\superblock.dd
Output: C:\Users\jack\Desktop\FAU.x64\superblock.dd
1024 bytes
1+0 records in
1+0 records out
1024 bytes written

Succeeded!
Copying C:\Users\jack\Desktop\FAU.x64\FCopy.dd:Zone.Identifier to C:\Users\jack\Desktop\FAU.x64\FCopy.dd_Zone.Identifier
Output: C:\Users\jack\Desktop\FAU.x64\FCopy.dd_Zone.Identifier
0 bytes
0+0 records in
0+0 records out
0 bytes written

Succeeded!
```

Figure 17 Extraction of superblock

Here the contents of the superblock have been ported to a file called ‘superblock.dd’, I then opened the file using winhex and began to analyse the superblock contents.



```
  Offset     0  1  2  3  4  5  6  7   8  9 10 11 12 13 14 15
 00000000   00 F1 00 00 00 C3 03 00  26 30 00 00 9F A2 03 00    ñ   Ã  &0  Ÿ¢
 00000016   F5 F0 00 00 00 00 00 00  02 00 00 00 02 00 00 00   õð
 00000032   00 80 00 00 00 80 00 00  20 1E 00 00 28 CC 8D 4F    €   €      (Ì O
 00000048   28 CC 8D 4F 01 00 22 00  53 EF 01 00 01 00 00 00   (Ì O  " Sï
 00000064   71 CB 8D 4F 00 4E ED 00  00 00 00 00 01 00 00 00   qË O Ní
 00000080   00 00 00 00 0B 00 00 00  00 01 00 00 3C 00 00 00               <
 00000096   06 00 00 00 03 00 00 00  3F BC 0D 46 7C 6B 46 28           ?¼ F|kF(
 00000112   92 08 B6 35 32 34 AF AD  41 4F 42 72 69 65 6E 20   ’ ¶524¯-AOBrien
 00000128   55 53 42 20 53 74 69 63  00 00 00 00 00 00 00 00   USB Stic
 00000144   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
 00000160   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
 00000176   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
 00000192   00 00 00 00 00 00 00 00  00 00 00 00 00 00 3C 00                 <
 00000208   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
```

Figure 18 Winhex Decoding

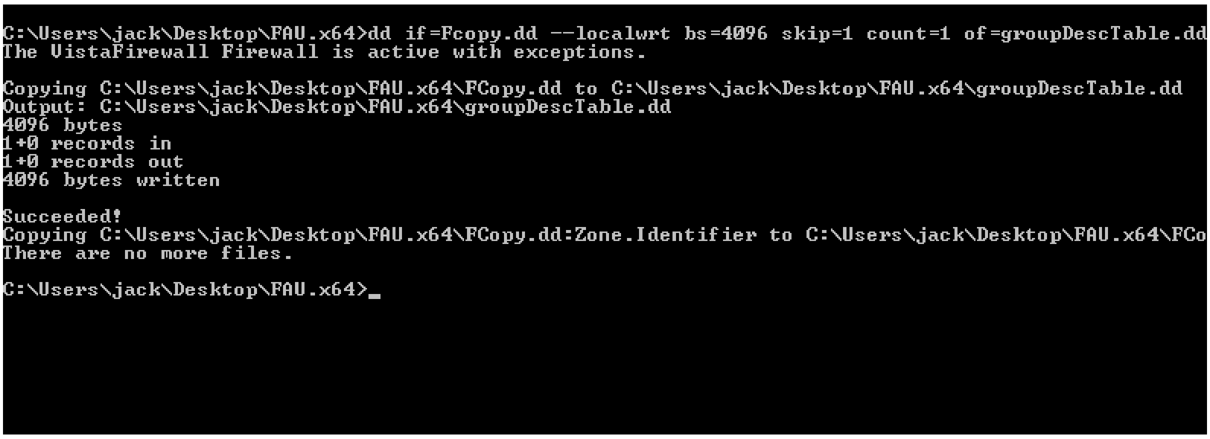
[2] http://www.slashroot.in/understanding-file-system-superblock-linux Superblocks

Sample of Superblock

Byte Range	Description	Size
0-3	Number of inodes in the file system.	61,696
4-7	Number of blocks in the file system.	246,528
8-11	Number of blocks reserved to prevent file system from filling up.	12,236
12-150	Number of unallocated blocks.	238,239

Group Descriptor Table

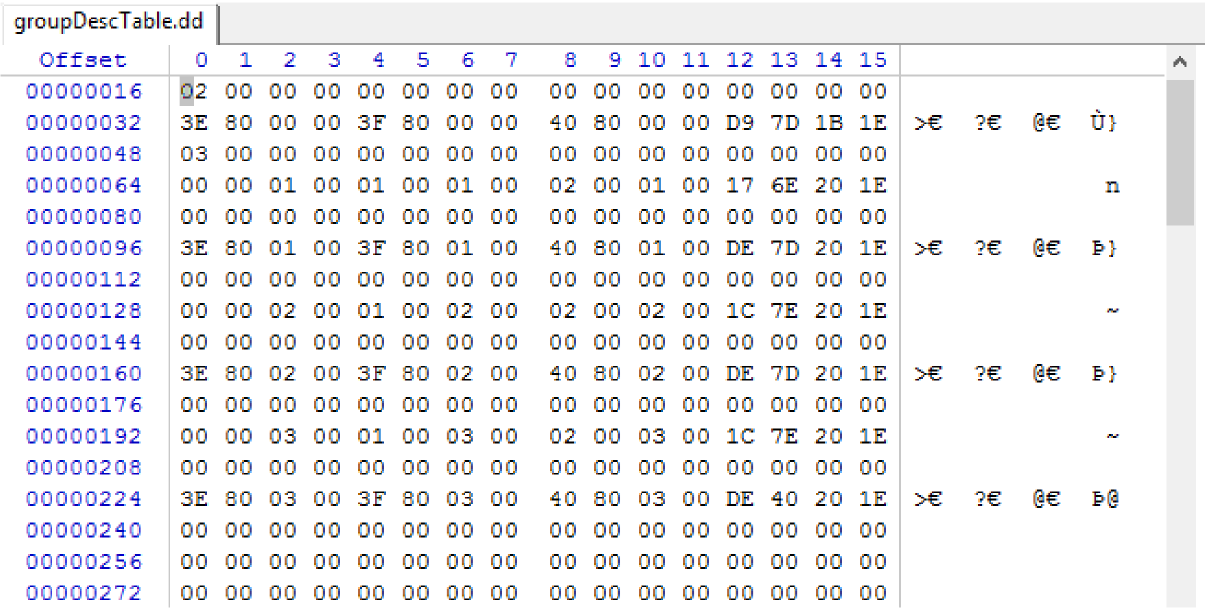
To extract the group descriptor table from the ext3 file system I used the FAU commands to dd out a copy of the table called ‘groupDescTable.dd’.



```
C:\Users\jack\Desktop\FAU.x64>dd if=Fcopy.dd --localwrt bs=4096 skip=1 count=1 of=groupDescTable.dd
The VistaFirewall Firewall is active with exceptions.

Copying C:\Users\jack\Desktop\FAU.x64\FCopy.dd to C:\Users\jack\Desktop\FAU.x64\groupDescTable.dd
Output: C:\Users\jack\Desktop\FAU.x64\groupDescTable.dd
4096 bytes
1+0 records in
1+0 records out
4096 bytes written

Succeeded!
Copying C:\Users\jack\Desktop\FAU.x64\FCopy.dd:Zone.Identifier to C:\Users\jack\Desktop\FAU.x64\FCo
There are no more files.

C:\Users\jack\Desktop\FAU.x64>_
```

Figure 19 Extracting group descriptor table

I then decoded the hexadecimal of the table using Winhex.



```
groupDescTable.dd

  Offset    0  1  2  3  4  5  6  7   8  9 10 11 12 13 14 15
00000016   02 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000032   3E 80 00 00 3F 80 00 00  40 80 00 00 D9 7D 1B 1E  >€  ?€  @€  Ù}
00000048   03 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000064   00 00 01 00 01 00 01 00  02 00 01 00 17 6E 20 1E              n
00000080   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000096   3E 80 01 00 3F 80 01 00  40 80 01 00 DE 7D 20 1E  >€  ?€  @€  Þ}
00000112   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000128   00 00 02 00 01 00 02 00  02 00 02 00 1C 7E 20 1E              ~
00000144   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000160   3E 80 02 00 3F 80 02 00  40 80 02 00 DE 7D 20 1E  >€  ?€  @€  Þ}
00000176   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000192   00 00 03 00 01 00 03 00  02 00 03 00 1C 7E 20 1E              ~
00000208   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000224   3E 80 03 00 3F 80 03 00  40 80 03 00 DE 40 20 1E  >€  ?€  @€  Þ@
00000240   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000256   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000272   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
```

Figure 20 Winhex Hex

Sample of Group Descriptor Table

Byte Range	Description	Size
0-3	Starting block address of block bitmap.	62
4-7	Starting bock address of inode bitmap.	63
8-11	Starting block address of inode table.	64
12-13	Number of unallocated blocks in group.	31,364
14-15	Number of unallocated inodes in group.	7,692
16-17	Number of directories in group.	2

Summary

This report has looked at the use and contents of the USB stick owned by former KDM employee Anne ’O Brien, there has been multiple PDFs of patented information retrieved from the USB, also including two deleted PDFs that had been tampered with and were temporarily deleted. All of the PDFs have been extracted safely from the USB using controlled copying practices and the original file has been left untouched.

Bibliography

http://www.thegeekstuff.com/2011/05/ext2-ext3-ext4 Date Accessed: 14/03/2017,

By Ramesh Natarajan on May 16, 2011. Ext3 statistics/information.

http://www.slashroot.in/understanding-file-system-superblock-linux Date Accessed: 18/03/2017, By Sarath Pillai on July 9, 2015. Superblock explanation

Carrier, Brian, File System Forensic Analysis, March 17, 2005