



Digital Discovery with Linux Bootable CDs

Dr. Philip Craiger, CISSP

Assistant Director for Digital Evidence

National Center for Forensic Science

&

Assistant Professor

Department of Engineering Technology

University of Central Florida

Linux for First Responders & Forensics

- Advantages
 - Open source, cheap, readily available
 - Many Linux distributions are freely downloadable
 - Many choices for bootable Linux CDs
 - Doesn't automount partitions read-write
 - Can mount dozens of different file systems
 - VFAT, NTFS, EXT2/3, HPFS, FFS
 - 90% of all computers run some form of Windows, can do numerous types of analysis, including recovering data from the registry!
 - Forensic toolkit on a single CD

man mount

-t vfstype

The argument following the **-t** is used to indicate the file system type. The file system types which are currently supported are: adfs, affs, autofs, coda, coherent, cramfs, devpts, efs, ext, ext2, ext3, hfs, hpfs, iso9660, jfs, minix, msdos, ncpfs, nfs, ntfs, proc, qnx4, ramfs, reiserfs, romfs, smbfs, sysv, tmpfs, udf, ufs, umsdos, vfat, xenix, xfs, xiafs. Note that coherent, sysv and xenix are equivalent and that xenix and coherent will be removed at some point in the future – use sysv instead. Since kernel version 2.1.21 the types ext and xiafs do not exist anymore.

The Linux Alternative

- Disadvantages
 - Requires more technical knowledge
 - NOT user friendly
 - Same procedures can be much more time consuming
 - Not for GUI-jockeys
 - Has not been subjected to as much scrutiny in court cases as COTS
 - However, dd (GNU & FreeBSD) tested by NIST
 - Some COTS tested in court cases
 - e.g., EnCase, probably ILook

Sound Cyberforensic Procedures

- Three steps
 - *Acquire* the evidence
 - Make a *forensically* sound image, OR
 - Preview the evidence without mounting disk or otherwise altering evidence
 - *Authenticate* the evidence
 - Verify the integrity of the copy
 - *Analyze* the evidence
 - Logical analysis: From a file system viewpoint
 - Physical analysis: From a flat file viewpoint
- Linux supports both logical & physical analysis

EVERYTHING is a file...

- Everything is a file
 - **Every** physical device will be associated with a file in the /dev directory
 - /dev/hda = first IDE hard drive on **primary** controller
 - /dev/hda1 = first partition on the drive
 - /dev/hdc = first IDE hard drive on the **secondary** controller
 - /dev/hdc5 = fifth partition on the drive
 - /dev/sda = first SCSI hard drive
 - /dev/sda7 = seventh partition on the drive
 - USB devices typically show up as /dev/sda

Acquire the Evidence

- How to create a 'forensically sound image'
 - Make an 'exact,' bit-for-bit, physical duplicate
 - dd is a utility available with all Linux/UNIX distributions
- # dd if=/dev/fd0 of=floppy.dd
 - '*if*' is the source of the bits
 - '*of*' is the name of the file to be written
- How do I know the # drives? The # of partitions? The file systems involved?
 - # fdisk -l
 - fdisk is a utility used for partitioning. NOT the same as fdisk that comes with DOS

Determine drive geometry with fdisk -l

```
root@gheera:~  
File Edit View Terminal Go Help  
[root@gheera root]# fdisk -l
```

Disk /dev/hda: 40.0 GB, 40016019456 bytes
16 heads, 63 sectors/track, 77536 cylinders
Units = cylinders of 1008 * 512 = 516096 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/hda1	*	1	24385	12289693+	7	HPFS/NTFS
/dev/hda2		24385	40641	8193150	7	HPFS/NTFS
/dev/hda3		40642	40844	102312	83	Linux
/dev/hda4		40845	77536	18492768	f	Win95 Ext'd (LBA)
/dev/hda5		40845	75456	17444416+	83	Linux
/dev/hda6		75457	77536	1048288+	82	Linux swap

Disk /dev/hdb: 200.0 GB, 200049647616 bytes
255 heads, 63 sectors/track, 24321 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/hdb1	*	1	11475	92172906	7	HPFS/NTFS
/dev/hdb3		11476	16709	42042105	c	Win95 FAT32 (LBA)
/dev/hdb4		16710	24321	61143390	f	Win95 Ext'd (LBA)
/dev/hdb5		16710	24321	61143358+	7	HPFS/NTFS

```
[root@gheera root]#
```

Drive geometry

Disk /dev/hda: 40.0 GB, 40016019456 bytes
16 heads, 63 sectors/track, 77536 cylinders
Units = cylinders of 1008 * 512 = 516096 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/hda1	*	1	24385	12289693+	7	HPFS/NTFS
/dev/hda2		24385	40641	8193150	7	HPFS/NTFS
/dev/hda3		40642	40844	102312	83	Linux
/dev/hda4		40845	77536	18492768	f	Win95 Ext'd (LBA)
/dev/hda5		40845	75456	17444416+	83	Linux
/dev/hda6		75457	77536	1048288+	82	Linux swap

Different Methods of Acquiring Images

- Different types each with different advantages & disadvantages
 - Direct connection
 - Connect drive directly to IDE ribbon cable
 - Fast
 - External HD case
 - Fast if used with Firewire
 - Network acquisition
 - Connect computer with cross-over cable

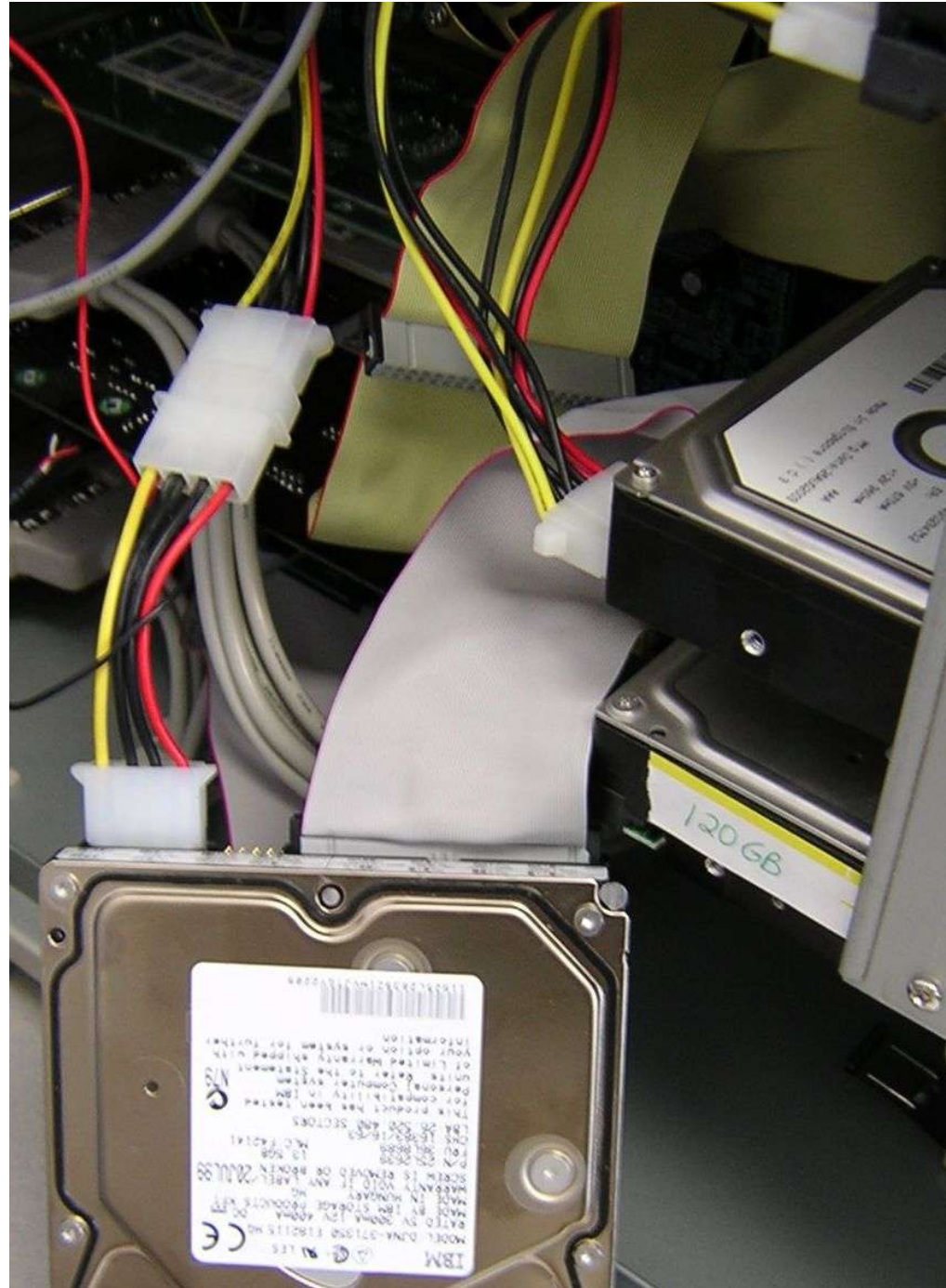
Firewire with Write-Blocker

Write-block is necessary for imaging under Windows



Direct IDE Connection

No write-blocker
necessary if imaging
under Linux



External Firewire Drive Connector



Network/Crossover Cable Acquisition

- Boot client with Linux bootable CD
- Setup network connection between forensic server and client.
- Send the information over the network connection
 - This can also be done over the Internet, i.e., from a distance.
 - However, care must be taken that no one eavesdrops.



I'm sending bits on port 9999

I'm listening on port 9999

Types of Analysis

- Logical
 - From a file system viewpoint
 - Folders/directories, files, etc.
 - View files, contents, capture metadata
- Physical
 - From a flat file viewpoint
 - There are no 'files' or 'folders'
 - Can view hidden systems areas
 - MBR, VBR, FAT, MFT, root directory, slack, unallocated, etc.

Logical View

The screenshot shows the Konqueror file manager window. The title bar reads "file:/home/jpc/docs/portable.linux/floppy - Konqueror". The menu bar includes "Location", "Edit", "View", "Go", "Bookmarks", "Tools", "Settings", "Window", and "Help". The location bar shows "file:/home/jpc/docs/portable.linux/floppy". The left sidebar shows a tree view of the file system, with "portable.linux/floppy" selected. The main pane displays a table of files with columns for Name, Size, File Type, Modified, and Permissions.

Name	Size	File Type	Modified	Permissions
0392.txt	31.4 KB	Plain Text	2004-03-03 11:48	rw-r-xr-x
forensics.rtf	8.1 KB	Rich Text Format	2004-03-03 11:42	rw-r-xr-x
homework.doc	84.2 KB	PNG Image	2004-03-03 11:47	rw-r-xr-x
outlook.pst.recovery	2.7 KB	Plain Text	2004-03-03 11:42	rw-r-xr-x
snort.snarf.txt	1.5 KB	Plain Text	2004-03-03 11:42	rw-r-xr-x

5 Items - 5 Files (127.9 KB Total) - 0 Directories

Physical View

```
jpc@simba:~/docs/portable.linux - Shell - Konsole
Session Edit View Bookmarks Settings Help
0000000: eb3c 904d 5344 4f53 352e 3000 0201 0100  .<.MSDOS5.0.....
0000010: 02e0 0040 0bf0 0900 1200 0200 0000 0000  ...@.....
0000020: 0000 0000 0000 29d3 7332 544e 4f20 4e41  .....).s2TNO NA
0000030: 4d45 2020 2020 4641 5431 3220 2020 33c9  ME    FAT12  3.
0000040: 8ed1 bcf0 7b8e d9b8 0020 8ec0 fcbd 007c  ....{.... |
0000050: 384e 247d 248b c199 e83c 0172 1c83 eb3a  8N$}$....<.r...:
0000060: 66a1 1c7c 2666 3b07 268a 57fc 7506 80ca  f..|&f;.&.W.u...
0000070: 0288 5602 80c3 1073 eb33 c98a 4610 98f7  ..V....s.3..F...
0000080: 6616 0346 1c13 561e 0346 0e13 d18b 7611  f..F..V..F....v.
0000090: 6089 46fc 8956 feb8 2000 f7e6 8b5e 0b03  `..F..V.. ^..
00000a0: c348 f7f3 0146 fc11 4efe 61bf 0000 e8e6  .H...F..N.a....
00000b0: 0072 3926 382d 7417 60b1 0bbe a17d f3a6  .r9&8-t.`....}..
00000c0: 6174 324e 7409 83c7 203b fb72 e6eb dca0  at2Nt... ;.r....
: █
```

Logical Analysis

- Searching for files in **allocated** space
 - Viewing the files contents
- Finding known files
 - Known good files: Windows system files, office files, etc.
 - Remove these from analyses
 - Known bad files: Hacking tools, child exploitation, etc.
- Use MD5s of either bad or good known files to compare to files on the disk.
- Can be done; however, **MUCH** easier using GUI tools.

Finding files

- By
 - Last accessed
 - Last modified
 - Date/time created/changed
 - Type (file, directory, socket, etc.)
 - Permissions
 - Owner
 - File type
 - Inode number
 - Yada yada yada

Finding files by type

- `# find / -type f -name '*.doc'`
 - All files that are Word documents (end in *.doc extension)
- `# find / \(-name '*.doc' -o -name '*.xls' -o -name '*.ppt' \)`
 - All Office documents
 - Powerpoint OR Word OR Excel, etc.
 - Only look at extensions: Not a good thing to do.
- `# find / ! \(-name '*.doc' -o -name '*.xls' -o -name '*.ppt' \)`
 - ?

Other searches

- Find phone numbers within any document with
 - # `egrep '\ (* [1-9] [0-9] [0-9] \) * [-] * [0-9] [0-9] [0-9] [-] * [0-9] [0-9] [0-9] '` *
 - Can it find?
 - 8005551212
 - (800) 555-1212
 - (800)5551212
 - 800-555-1212
 - 800 - 555 - 1212

Physical Analysis

- Finding information in unallocated space
 - Deleted files
 - Swap space
 - Part of a previous file partially overwritten, partially there.



I want to kill americans. I am going to blow up buildings and kill all.



Dearest Friend: I am glad to hear from you. buildings and kill all.

What happens when a file is deleted

- In FAT
 - Root directory entry holds the starting cluster
 - FAT is a singly-linked list of the clusters comprising the file
- When a file is deleted, two things happen
 - The root directory entry is marked as available (e5)
 - The FAT entries are zeroed out, indicating they are available for us
- What **doesn't** happen
 - The information comprising the file is not removed, remains until overwritten
 - The root directory entry remains until overwritten

How to manually recover a file

- Find the starting cluster
- Find the file size
- Extract the contents of the image starting at the starting cluster
- We can do this with *dd*

```
jpc@tiger:~ - Shell - Konsole
Session Edit View Bookmarks Settings Help
0002800: e533 3932 2020 2020 4a50 4720 1000 8d5d .392 JPG ...]
0002810: 6330 6330 0000 8d5d 6330 7f01 a07d 0000 c0c0...]c0...}..
0002820: 4168 006f 006d 0065 0077 000f 006a 6f00 An.o.m.e.w...jo.
0002830: 7200 6b00 2e00 6400 6f00 0000 6300 0000 r.k...d.o...c...
0002840: 484f 4d45 574f 524b 444f 4320 1800 e45d HOMEWORKDOC ...]
0002850: 6330 6330 0000 e45d 6330 be01 e550 0100 c0c0...]c0...P..
0002860: 4130 0033 0039 0032 002e 000f 00df 7400 A0.3.9.2.....t.
0002870: 7800 7400 0000 ffff ffff 0000 ffff ffff x.t.....
0002880: 3033 3932 2020 2020 5458 5420 1000 155e 0392 TXT ...^
0002890: 6330 6330 0000 155e 6330 6702 a07d 0000 c0c0...^c0g..}..
00028a0: e530 0035 0031 0039 002e 000f 0051 6a00 .0.5.1.9.....qj.
00028b0: 7000 6700 0000 ffff ffff 0000 ffff ffff p.g.....
00028c0: e535 3139 2020 2020 4a50 4720 1000 8d5d .519 JPG ...]
00028d0: 6330 6330 0000 8d5d 6330 fe01 d079 0000 c0c0...]c0...y..
00028e0: e530 0035 0032 0030 002e 000f 00a2 6a00 .0.5.2.0.....j.
00028f0: 7000 6700 0000 ffff ffff 0000 ffff ffff p.g.....
0002900: e535 3230 2020 2020 4a50 4720 1000 8d5d .520 JPG ...]
0002910: 6330 6330 0000 8d5d 6330 3b02 a672 0000 c0c0...]c0;..r..
0002920: e530 0035 0032 0031 002e 000f 00a4 6a00 .0.5.2.1.....j.
0002930: 7000 6700 0000 ffff ffff 0000 ffff ffff p.g.....
0002940: e535 3231 2020 2020 4a50 4720 1000 8d5d .521 JPG ...]
0002950: 6330 6330 0000 8d5d 6330 7502 7367 0000 c0c0...]c0u.sg..
0002960: e530 0035 0032 0032 002e 000f 0076 6a00 .0.5.2.2.....vj.
:
```

Root Directory Entry for ?392.JPG

```
0 4720 1000 8d5d .392 [JPG ...]  
0 7f01 a07d 0000 c0c0... ]c0... }..
```

Starting Size = 0xA07D
Cluster
0x7F01

JPG header = FFh D8h FFh E0h

```
jpc@tiger:~ - Shell - Konsole
Session Edit View Bookmarks Settings Help
0033c00: ffd8 ffe0 0010 4a46 4946 0001 0101 012c .....JFIF.....,
0033c10: 012c 0000 ffdb 0043 0003 0202 0302 0203 ..,.....C.....
0033c20: 0303 0304 0303 0405 0805 0504 0405 0a07 .....
0033c30: 0706 080c 0a0c 0c0b 0a0b 0b0d 0e12 100d .....
0033c40: 0e11 0e0b 0b10 1610 1113 1415 1515 0c0f .....
0033c50: 1718 1614 1812 1415 14ff db00 4301 0304 .....C...
0033c60: 0405 0405 0905 0509 140d 0b0d 1414 1414 .....
0033c70: 1414 1414 1414 1414 1414 1414 1414 1414 .....
0033c80: 1414 1414 1414 1414 1414 1414 1414 1414 .....
0033c90: 1414 1414 1414 1414 1414 1414 1414 ffc0 .....
0033ca0: 0011 0801 4a01 7903 0122 0002 1101 0311 ....J.y.".....
0033cb0: 01ff c400 1e00 0100 0202 0301 0101 0000 .....
0033cc0: 0000 0000 0000 0005 0607 0804 090a 0301 .....
0033cd0: 02ff c400 6610 0000 0502 0401 030c 0a10 ....f.....
0033ce0: 0303 0807 0900 0102 0304 0500 0607 1112 .....
0033cf0: 1314 1521 2208 1617 2331 3253 5494 a3d2 ...!"...#12st...
0033d00: d418 3541 5657 7292 95b3 d324 2533 3436 ..5AVwr...$%346
0033d10: 4251 5255 5874 9396 a2b2 d161 6271 2643 BQRUXt....abq&C
0033d20: 7537 3865 7681 91b4 b509 6383 85b1 c1c2 u78ev....c.....
0033d30: 2746 6466 7384 86a1 f0ff c400 1b01 0100 'Fdfs.....
0033d40: 0203 0101 0000 0000 0000 0000 0000 0004 .....
0033d50: 0502 0306 0107 ffc4 0032 1101 0002 0102 .....2.....
0033d60: 0304 0709 0101 0000 0000 0000 0102 0304 .....
:█
```

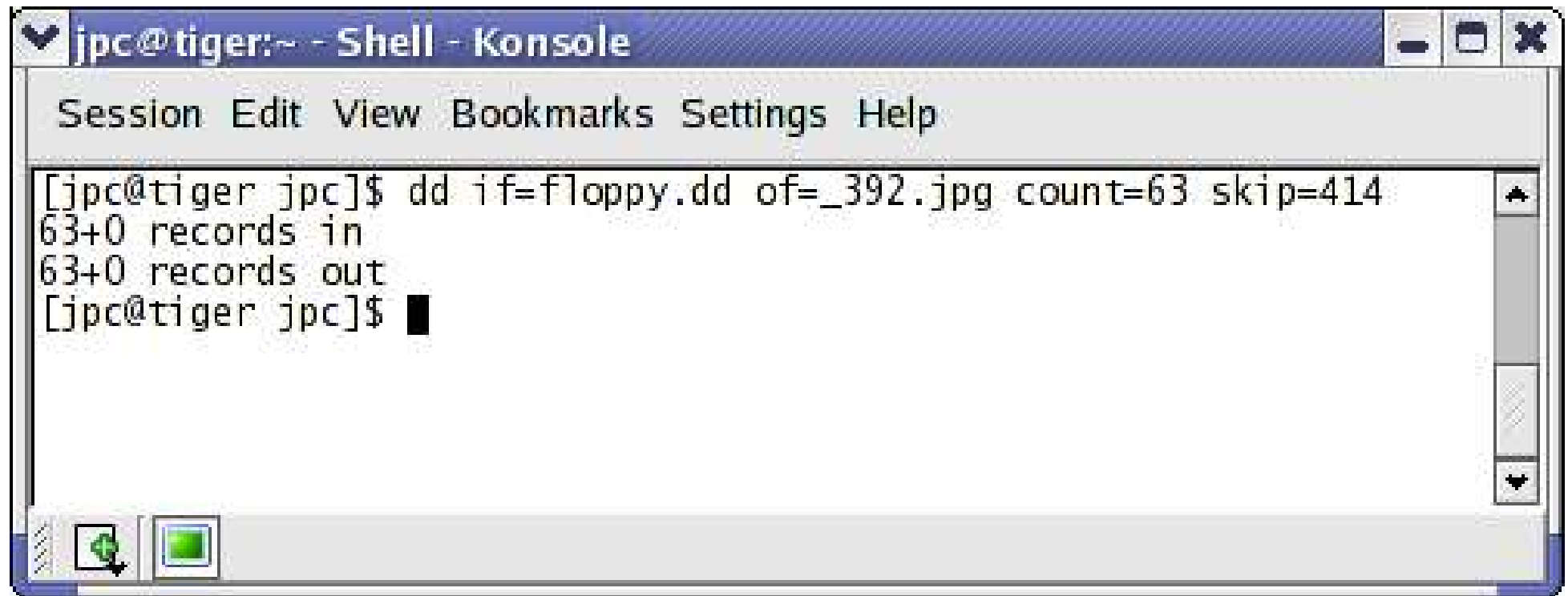
How to manually recover a file

- Procedure
 - Find the starting cluster
 - Find the file size
 - Extract the contents of the image starting at the starting cluster
- Example
 - $0x7F01 = \text{byteswap} = 0x17F$
 - $17Fh = 383$ decimal, the starting **logical** cluster
 - $383 + 31$ (reserved area) = 414 the **physical** cluster
 - 414×512 (sector size) = 211,968 = 33C00 byte offset

Manual File Recovery

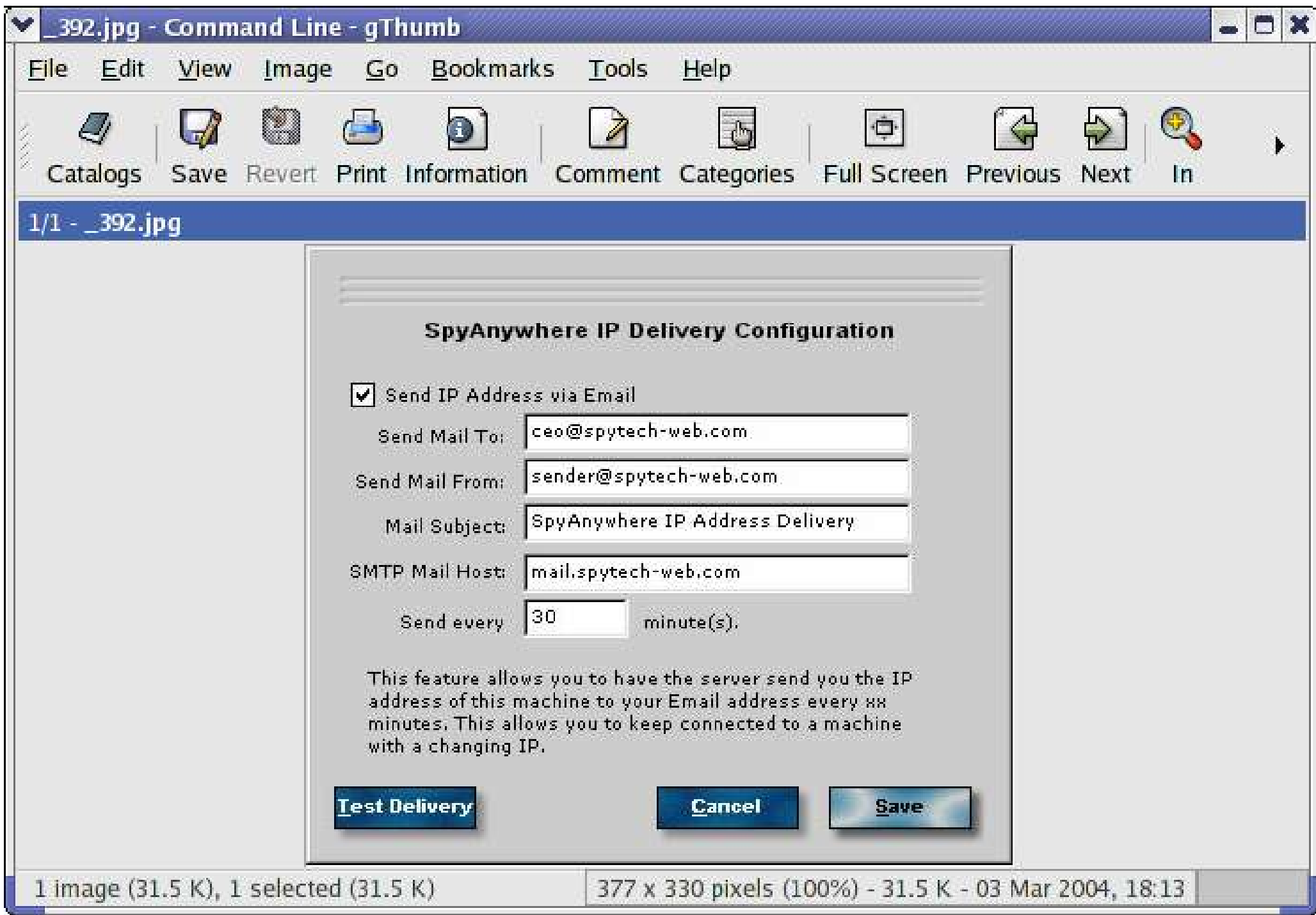
- Calculate file size
 - Byteswap $0xA07D = 0x7DA0$
 - $0x7DA0 = 32,160$ decimal
 - $32,160 / 512 = 62.8$ or 63 physical clusters
- ```
dd if=floppy.dd of=_392.jpg
skip=414 count=63
```

  - `count` = how many data blocks to grab (based on `dd`'s default size of 512!)
  - `skip` = the physical sector on which to begin recovering



A terminal window titled "jpc@tiger:~ - Shell - Konsole" with a menu bar containing "Session Edit View Bookmarks Settings Help". The terminal output shows a successful execution of the command `dd if=floppy.dd of=_392.jpg count=63 skip=414`. The output indicates that 63+0 records were read from the input file and 63+0 records were written to the output file. The prompt `[jpc@tiger jpc]$` is shown again with a cursor.

```
[jpc@tiger jpc]$ dd if=floppy.dd of=_392.jpg count=63 skip=414
63+0 records in
63+0 records out
[jpc@tiger jpc]$
```



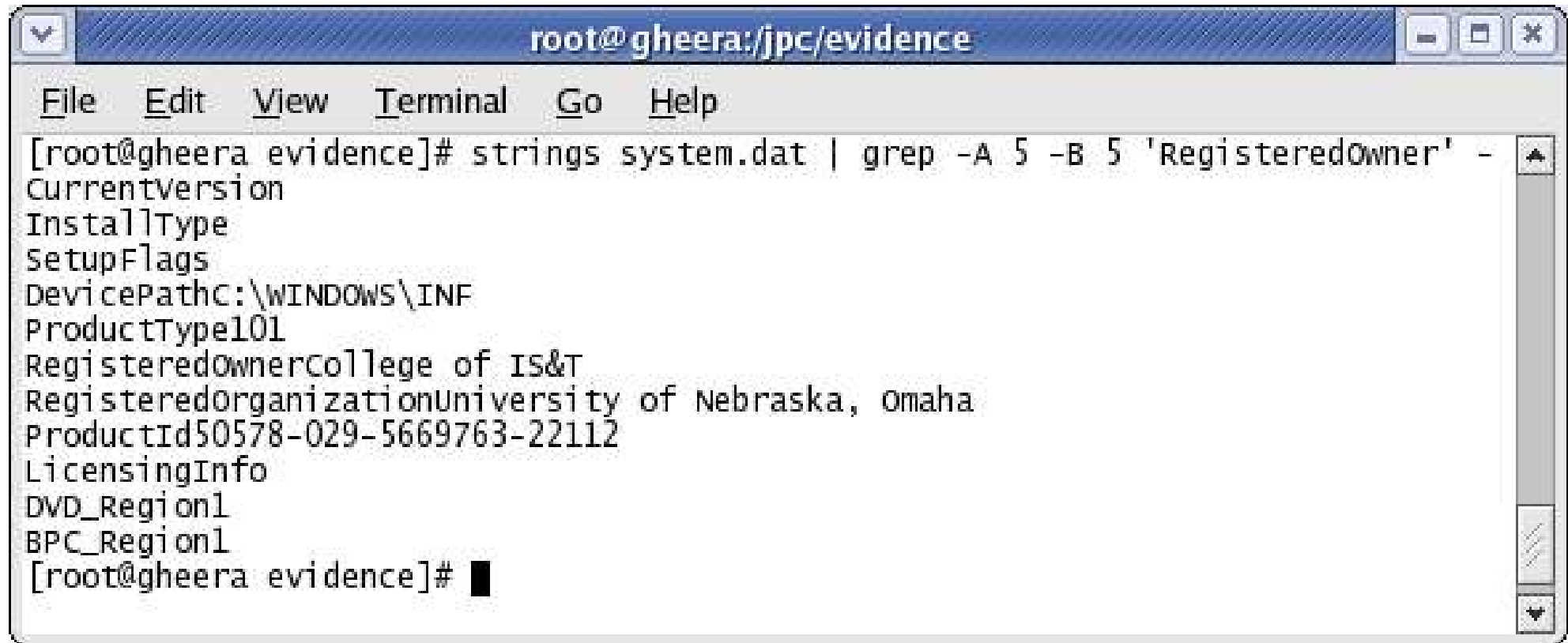
# Dealing with the Windows Registry

---

- What is the key name?
- Procedure for extracting information
  - Extract human readable strings
  - Search through the extracted strings for the expression
- Interesting registry entries
  - Registered Owner
  - Date installed
  - IP address
  - Software keys

# Registered Owner of this Computer?

---



A terminal window titled "root@gheera:/jpc/evidence" with a menu bar (File, Edit, View, Terminal, Go, Help). The terminal shows the command `strings system.dat | grep -A 5 -B 5 'RegisteredOwner'` and its output:

```
[root@gheera evidence]# strings system.dat | grep -A 5 -B 5 'RegisteredOwner' -
CurrentVersion
InstallType
SetupFlags
DevicePathC:\WINDOWS\INF
ProductType101
RegisteredOwnerCollege of IS&T
RegisteredOrganizationUniversity of Nebraska, Omaha
ProductId50578-029-5669763-22112
LicensingInfo
DVD_Region1
BPC_Region1
[root@gheera evidence]#
```

# Bootable Linux CDs

---

- Forensic Bootable (in order of my preference)
  - Helix: [www.e-fense.com](http://www.e-fense.com)
  - Knoppix-STD ([www.knoppix-std.org](http://www.knoppix-std.org))
  - Penguin Sleuth Kit ([www.linux-forensics.com](http://www.linux-forensics.com))
  - Local Area Security ([www.localareasecurity.com](http://www.localareasecurity.com))
- Non Forensic
  - Knoppix ([www.knoppix.com](http://www.knoppix.com))
  - Gnoppix ([www.gnoppix.org](http://www.gnoppix.org))
  - Several others

# Disadvantages

---

- GUI tools, including EnCase & FTK, automatically parse the following
  - Slack space
    - Space between the end of the file and the beginning of the next cluster
    - Information from previously allocated files
    - Can hold a great deal of information depending upon file size
  - Unallocated space
    - Space available for files
    - Includes deleted-not-as-yet-overwritten files

# References

---

- Craiger, J.P. (in press). Computer forensics procedures and methods. In H. Bigdoli (Ed.), *The Information Security Handbook*. John Wiley & Sons.
- Craiger, J.P. (in press). Recovering digital evidence from Linux systems. In S. Shenoit (Ed.), *Advances in Digital Forensics*. International Federation of Information Professionals.
- Craiger, J.P., Pollitt, M., & Swauger, J. (in press). Digital Evidence and Law Enforcement. To appear in H. Bigdoli (Ed.), *Handbook of Information Security*. Wiley & Sons.
- Craiger, J.P. (to appear). Digital Evidence Obfuscation: Recovery Techniques. To appear in Proceedings of the International Society for Optical Engineering.
- Swauger, J. & Craiger, J.P. (to appear). Digital Forensic Software Tool Validation. Submitted to P. Kanellis (Ed.), *Digital Crime and Forensic Science in Cyberspace*. Idea Group.