



Steganography, Steganalysis, & Cryptanalysis

Michael T. Raggio, CISSP
Principal Security Consultant
VeriSign

Agenda

- ▶ **Steganography**
 - **What is Steganography?**
 - **History**
 - **Steganography today**
 - **Steganography tools**
- ▶ **Steganalysis**
 - **What is Steganalysis?**
 - **Types of analysis**
 - **Identification of Steganographic files**
- ▶ **Steganalysis meets Cryptanalysis**
 - **Password Guessing**
 - **Cracking Steganography programs**
- ▶ **Forensics/Anti-Forensics**
- ▶ **Conclusions**
 - **What's in the Future?**
 - **Other tools in the wild**
 - **References**



Steganography

Steganography - Definition

- ▶ **Steganography**
 - from the Greek word **steganos** meaning “covered”
 - and the Greek word **graphie** meaning “writing”
- ▶ **Steganography is the process of hiding of a secret message within an ordinary message and extracting it at its destination**
- ▶ **Anyone else viewing the message will fail to know it contains hidden/encrypted data**



Steganography - History

- ▶ **Greek history – warning of invasion by scrawling it on the wood underneath a wax tablet. To casual observers, the tablet appeared blank.**
- ▶ **Both Axis and Allied spies during World War II used such measures as invisible inks -- using milk, fruit juice or urine which darken when heated.**
- ▶ **Invisible Ink is also a form of steganography**

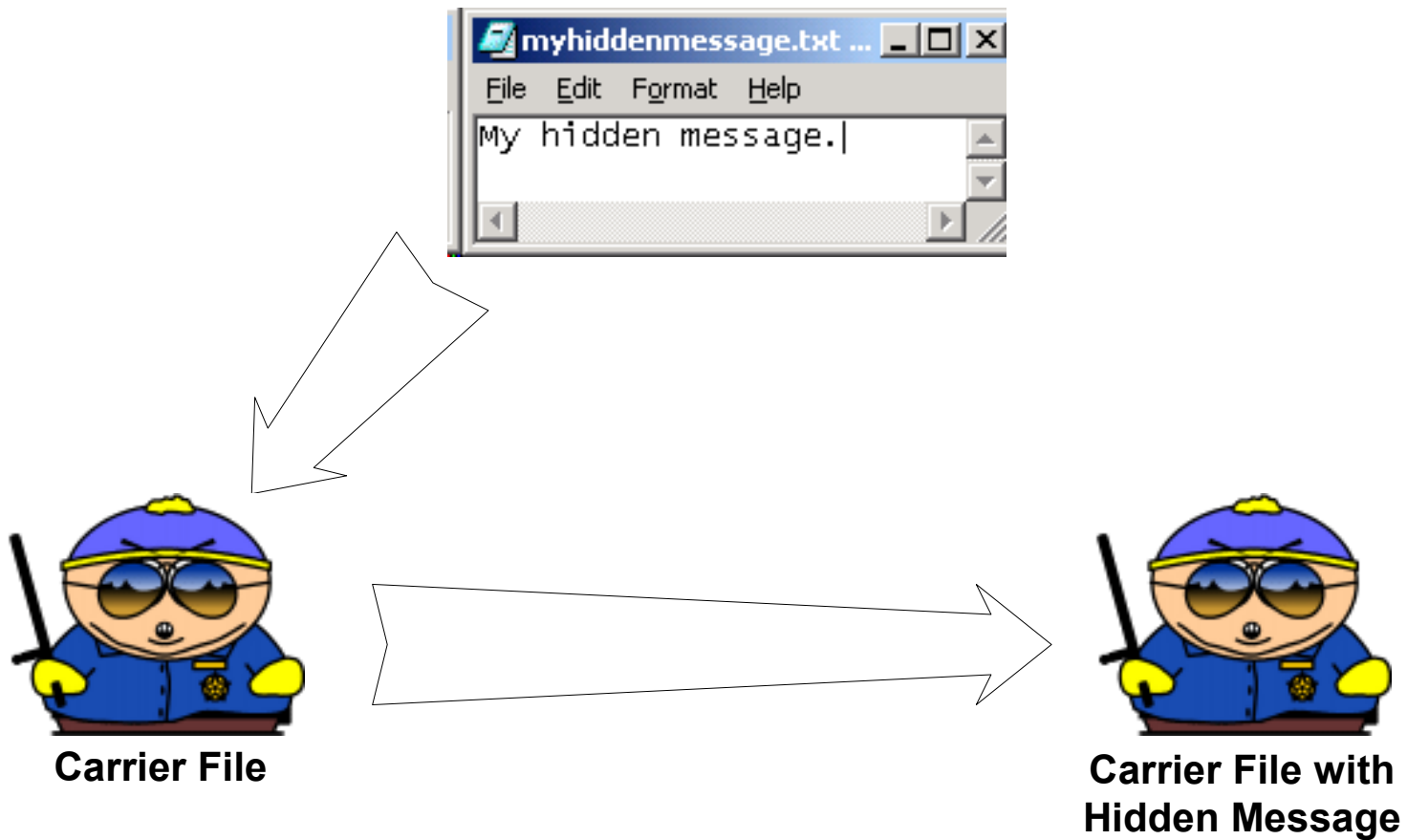
Steganography

- ▶ **The U.S. government is concerned about the use of Steganography.**
- ▶ **Common uses in include the disguising of corporate espionage.**
- ▶ **It's possible that terrorist cells may use it to secretly communicate information.**
 - This is rumored to be a common technique used by Al-Qaeda. By posting the image on a website for download by another terrorist cell. Using the same Steganography program, the terrorist cell could then reveal the message with plans for a new attack.
- ▶ **It's also a very good Anti-forensics mechanism to mitigate the effectiveness of a forensics investigation**
 - Child pornography

Steganography

- ▶ **Modern digital steganography**
 - data is encrypted
 - then inserted and hidden, using a special algorithm which may add and/or modify the contents of the file
 - This technique may simply append the data to the file, or disperse it throughout
 - Carefully crafted programs apply the encrypted data such that patterns appear normal.

Steganography – Modern Day



Steganography – Carrier Files

Steganography Carrier Files

- ▶ **bmp**
- ▶ **jpeg**
- ▶ **gif**
- ▶ **wav**
- ▶ **mp3**
- ▶ **Amongst others...**

Steganography - Tools

Steganography Tools

- ▶ **Steganos**
- ▶ **S-Tools (GIF, JPEG)**
- ▶ **StegHide (WAV, BMP)**
- ▶ **Invisible Secrets (JPEG)**
- ▶ **JPHide**
- ▶ **Camouflage**
- ▶ **Hiderman**
- ▶ **Many others...**

Steganography

▶ Popular sites for Steganography information

UPDATED URL:

<http://www.jjtc.com/neil/research.html>

- <http://www.rhetoric.umn.edu/Rhetoric/misc/dfrank/stegsoft.html> - No longer available site...

UPDATED URL:

<http://www.topology.org/soft/crypto.html>



Steganalysis

Identification of hidden files

Steganalysis - Definition

▶ **Definition**

- Identifying the existence of a message
- **Not** extracting the message
- Note: Technically, Steganography deals with the concealment of a message, not the encryption of it

▶ **Steganalysis essentially deals with the *detection* of hidden content**

▶ **How is this meaningful???**



Steganalysis

- ▶ **By identifying the existence of a hidden message, perhaps we can identify the tools used to hide it.**
- ▶ **If we identify the tool, perhaps we can use that tool to extract the original message.**



Steganalysis – Hiding Techniques

▶ Common hiding techniques

- Appended to a file
- Hidden in the unused header portion of the file near the beginning of the file contents
- An algorithm is used to disperse the hidden message throughout the file
 - ▶ Modification of LSB (Least Significant Bit)
 - ▶ Other



Steganalysis – Methods of Detection

- ▶ **Methods of detecting the use of Steganography**
 - Visual Detection (JPEG, BMP, GIF, etc.)
 - Audible Detection (WAV, MPEG, etc.)
 - Statistical Detection (changes in patterns of the pixels or LSB – Least Significant Bit) or Histogram Analysis
 - Structural Detection - View file properties/contents
 - ▶ size difference
 - ▶ date/time difference
 - ▶ contents – modifications
 - ▶ checksum

Steganalysis – Methods of Detection

▶ **Categories**

- Anomaly
 - ▶ Histogram analysis
 - ▶ Change in file properties
 - ▶ Statistical Attack
 - ▶ Visually
 - ▶ Audible
- Signature
 - ▶ A pattern consistent with the program used

Steganalysis – Methods of Detection

▶ **Goal**

- Accuracy
- Consistency
- Minimize false-positives



Anomaly – Visual Detection

- ▶ **Detecting Steganography by viewing it**

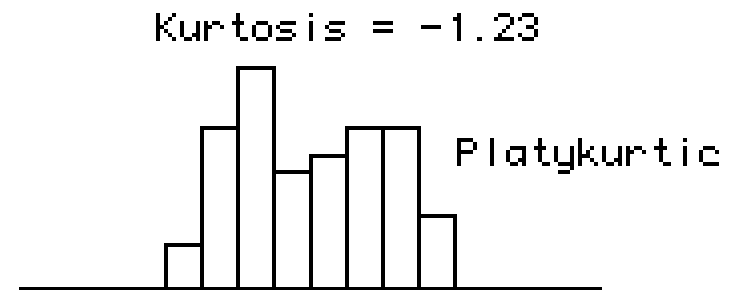
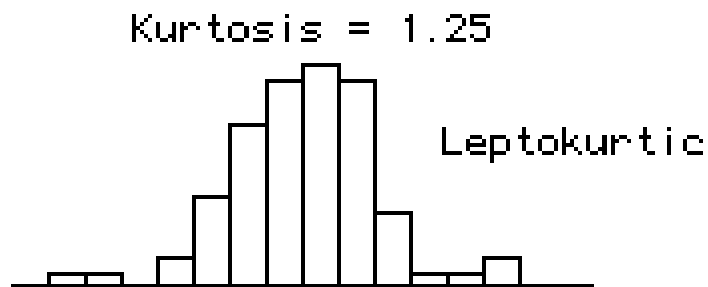


- ▶ **Can you see a difference in these two pictures?
(I can't!)**

Anomaly - Kurtosis

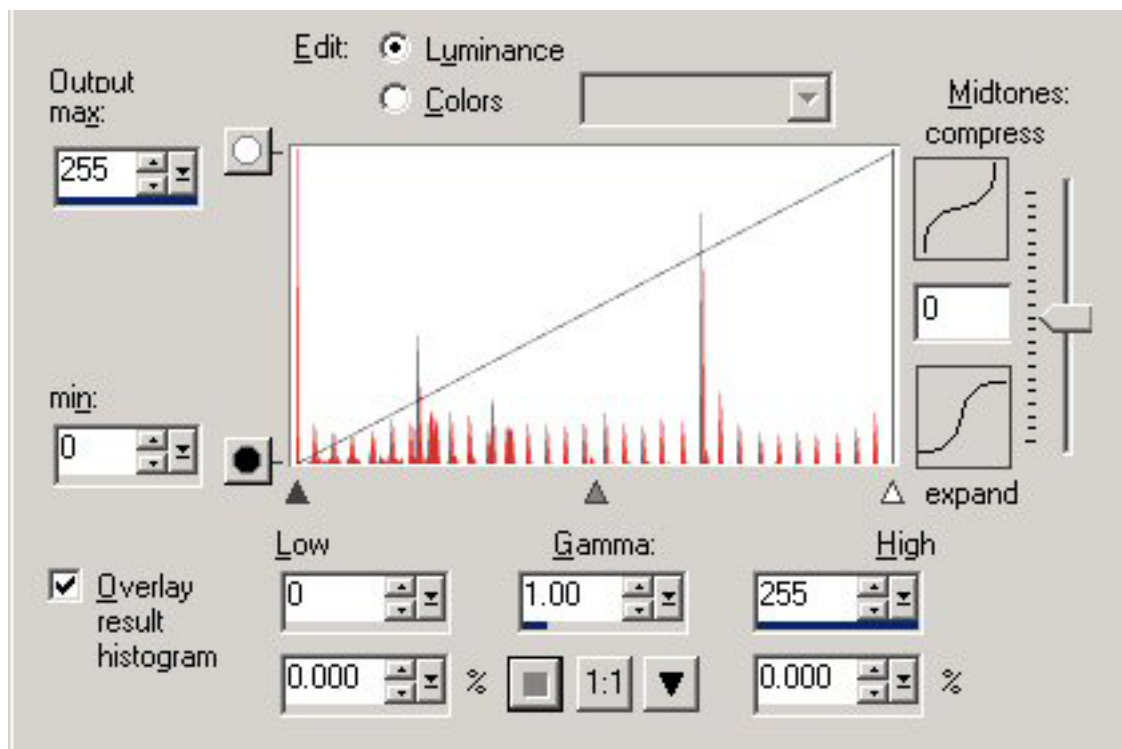
▶ Kurtosis

- The degree of flatness or peakedness of a curve describing a frequency of distribution
- Random House Dictionary



Anomaly – Histogram Analysis

- ▶ **By comparing histograms, we can see this histogram has a very noticeable repetitive trend.**



Anomaly Analysis - Compare file properties

- ▶ **Compare the properties of the files**



- ▶ **Properties**

- 04/04/2003 05:25p 240,759 helmetprototype.jpg
- 04/04/2003 05:26p 235,750 helmetprototype.jpg

- ▶ **Checksum**

- C:\GNUTools>cksum a:\before\helmetprototype.jpg
3241690497 240759 a:\before\helmetprototype.jpg
- C:\GNUTools>cksum a:\after\helmetprototype.jpg
3749290633 235750 a:\after\helmetprototype.jpg

File Signatures

HEX Signature	File Extension	ASCII Signature
FF D8 FF E0 xx xx 4A 46 49 46 00	JPEG (JPEG, JFIF, JPE, JPG)	ÿØÿà..JFIF.
47 49 46 38 37 61 47 49 46 38 39 61	GIF	GIF87a GIF89a
42 4D	BMP	BM

► For a full list see:

www.garykessler.net/library/file_sigs.html



Steganalysis – Analyzing contents of file

- ▶ **If you have a copy of the original (virgin) file, it can be compared to the modified suspect/carrier file**
- ▶ **Many tools can be used for viewing and comparing the contents of a hidden file.**
- ▶ **Everything from Notepad to a Hex Editor can be used to identify inconsistencies and patterns**
- ▶ **Reviewing multiple files may identify a signature pattern related to the Steganography program**



Steganalysis – Analyzing contents of file

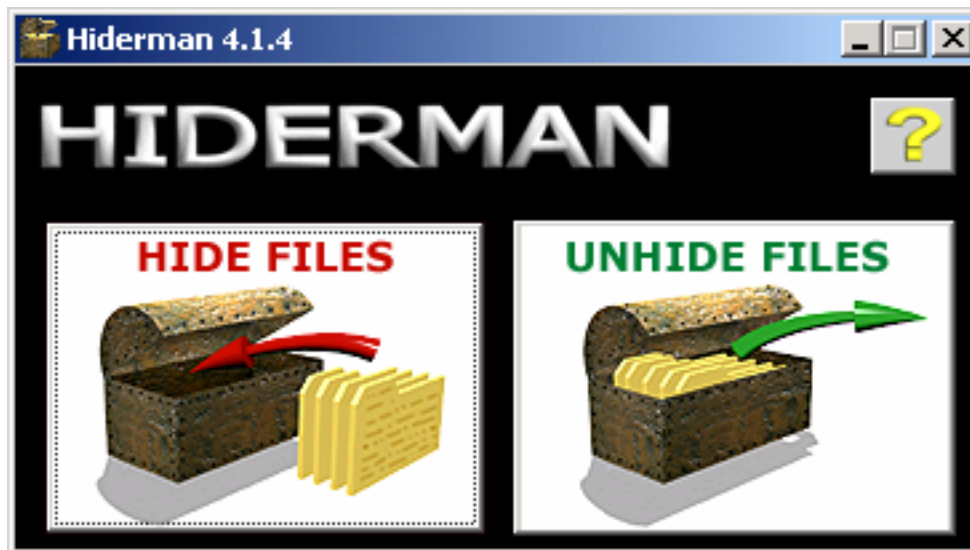
▶ Helpful analysis programs

- WinHex – www.winhex.com
 - ▶ Allows conversions between ASCII and Hex
 - ▶ Allows comparison of files
 - ▶ Save comparison as a report
 - ▶ Search differences or equal bytes
 - ▶ Contains file marker capabilities
 - ▶ Allows string searches – both ASCII and Hex
 - ▶ Many, many other features



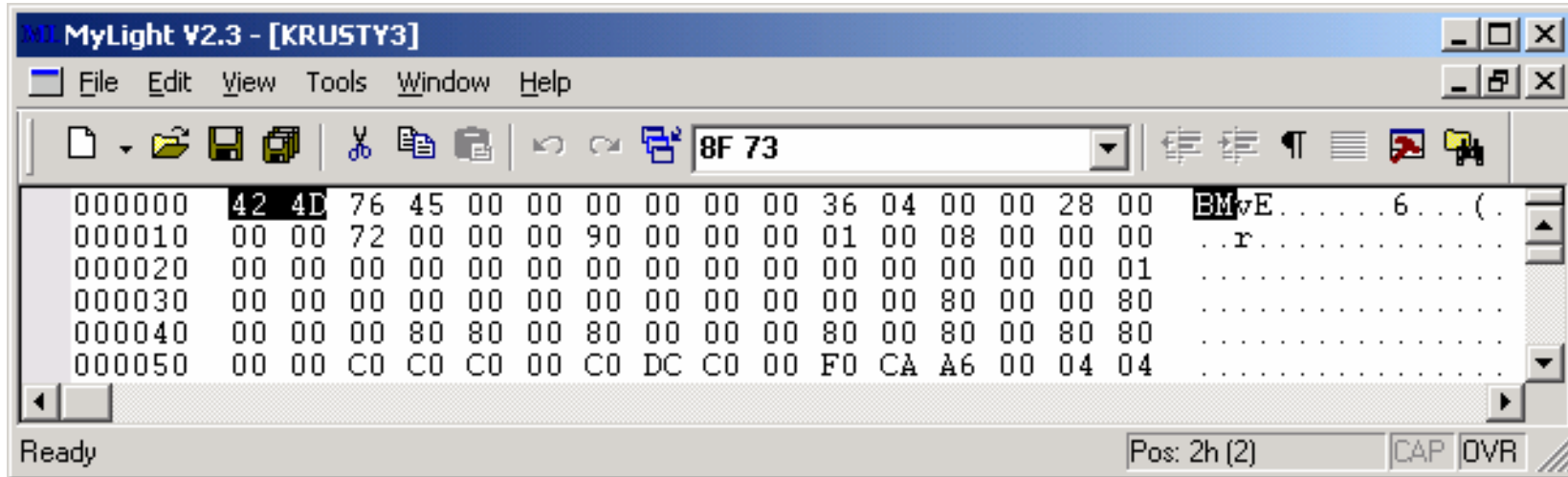
Hiderman – Case Study

- ▶ Let's examine a slightly sophisticated stego program – Hiderman



Hiderman – Case Study

- ▶ After hiding a message with Hiderman, we can review the file with our favorite Hex Tool.
- ▶ Viewing the Header information (beginning of the file) we see that it's a Bitmap as indicated by the "BM" file signature



The screenshot shows a hex editor window titled "MyLight V2.3 - [KRUSTY3]". The menu bar includes File, Edit, View, Tools, Window, and Help. The toolbar contains various icons for file operations and editing. The address bar shows "8F 73". The main display area shows a hex dump of the file's beginning. The first two bytes are highlighted in black and labeled "42 4D". The text column shows the signature "BM" followed by "vE.....6... (. .r.....".

Offset	Hex	ASCII
000000	42 4D 76 45 00 00 00 00 00 00 36 04 00 00 28 00	BMvE.....6... (.
000010	00 00 72 00 00 00 90 00 00 00 01 00 08 00 00 00	..r.....
000020	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01
000030	00 00 00 00 00 00 00 00 00 00 00 00 80 00 00 80
000040	00 00 00 80 80 00 80 00 00 00 80 00 80 00 80 80
000050	00 00 C0 C0 C0 00 C0 DC C0 00 F0 CA A6 00 04 04

Hiderman – Case Study

- ▶ **We then view the end of the file, comparing the virgin file to the carrier file**
- ▶ **Note the data appended to the file (on the next slide)**



Hiderman – Case Study

MyLight V2.3 - [KRUSTY3]

File Edit View Tools Window Help

8F 73

```
004520 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 .....
004530 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 .....
004540 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 .....
004550 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 .....
004560 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 .....
004570 01 01 01 01 00 00
```

3 byte(s) selected. Pos: 4573h (17779) CAP OVR

MyLight V2.3 - [KRUSTY3]

File Edit View Tools Window Help

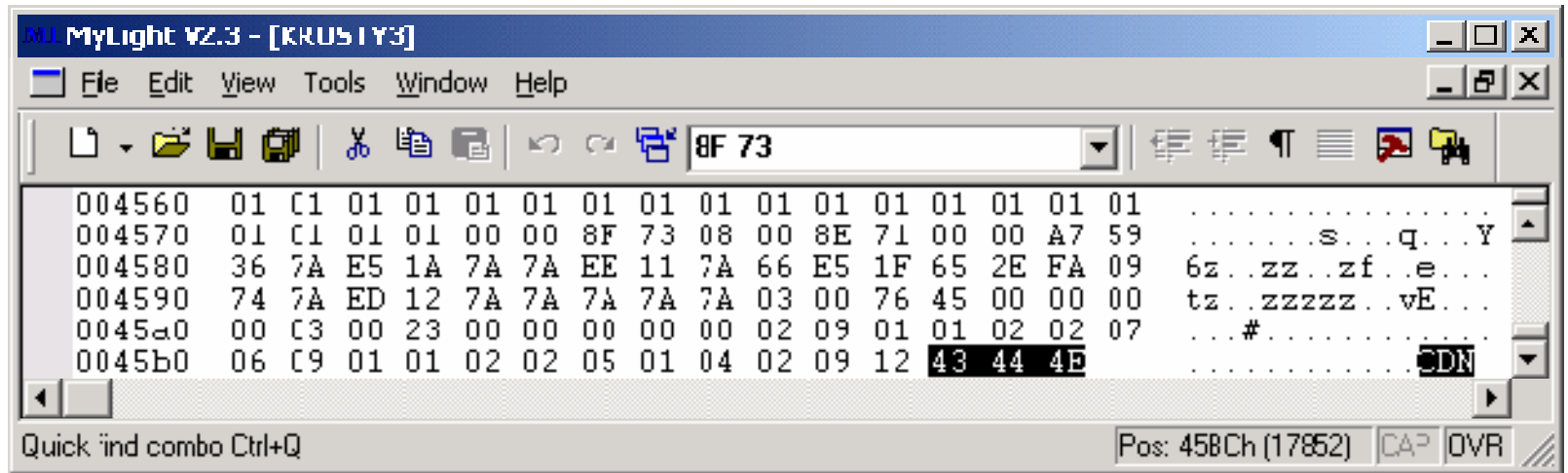
8F 73

```
004560 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 .....
004570 01 01 01 01 00 00 8F 73 08 00 8E 71 00 00 A7 59 s q Y
004580 36 7A E5 1A 7A 7A EE 11 7A 66 E5 1F 65 2E FA 09 6z .zz .zf .e
004590 74 7A ED 12 7A 7A 7A 7A 7A 03 00 76 45 00 00 00 tz .zzzzz .vE
0045a0 00 03 00 23 00 00 00 00 00 02 09 01 01 02 02 07 .#
0045b0 06 09 01 01 02 02 05 01 04 02 09 12 43 44 4E .....CDN
```

Quick find combo Ctrl+Q Pos: 4576h (17782) CAP OVR

Hiderman – Case Study

- ▶ In addition, note the last three characters “CDN” which is 43 44 4E in HEX.



```
004560 01 C1 01 01 01 01 01 01 01 01 01 01 01 01 01 01 .....
004570 01 C1 01 01 01 00 00 8F 73 08 00 8E 71 00 00 A7 59 .....s...q...Y
004580 36 7A E5 1A 7A 7A EE 11 7A 66 E5 1F 65 2E FA 09 6z..zz..zf..e...
004590 74 7A ED 12 7A 7A 7A 7A 7A 03 00 76 45 00 00 00 tz..zzzzz..vE...
0045a0 00 C3 00 23 00 00 00 00 00 02 09 01 01 02 02 07 ...#.....
0045b0 06 C9 01 01 02 02 05 01 04 02 09 12 43 44 4E .....CDN
```

Quick find combo Ctrl+Q Pos: 458Ch (17852) CA² OVR

Hiderman – Case Study

- ▶ **Hiding different messages in different files with different passwords, we see that the same three characters (“CDN”) are appended to the end of the file.**
- ▶ **Signature found.**

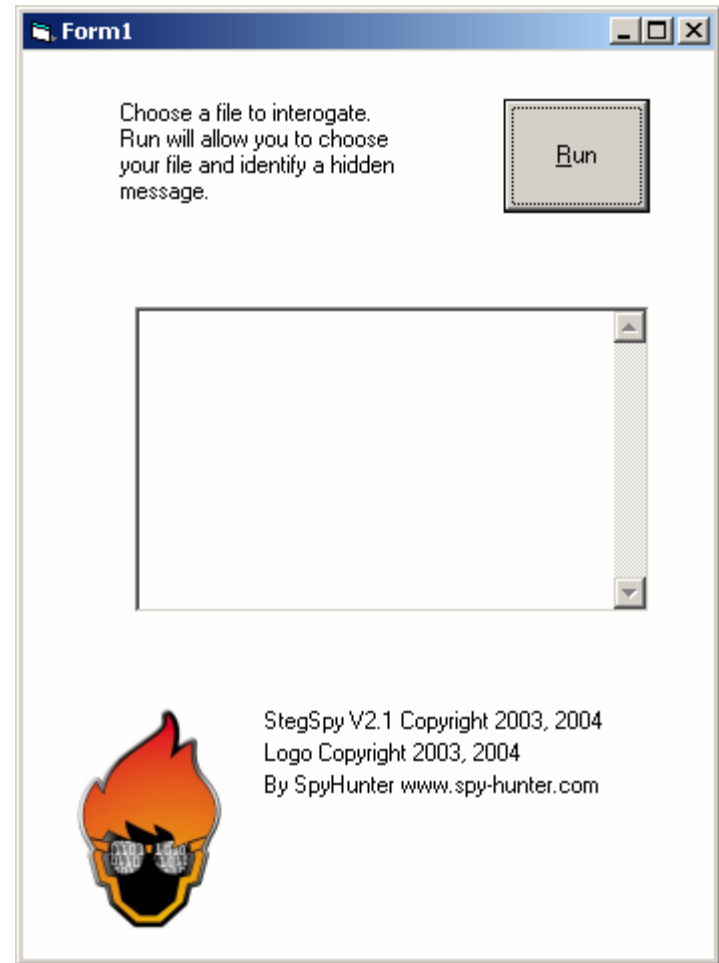
The screenshot shows the MyLight V2.3 hex editor interface. The main window displays a hex dump of a file. The address range is from 003890 to 0038f0. The hex data is as follows:

Address	Hex Data	ASCII Data
003890	24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24	\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
0038a0	24 24 24 24 24 24 24 24 24 24 24 24 24 24 24 24	\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$\$
0038b0	24 24 24 24 00 00 9C 60 04 00 9F 60 00 00 AF 50	\$\$\$\$.....`...`...P
0038c0	38 74 FA 13 74 2E EB 18 74 74 65 73 74 03 00 B6	8t...t...ttest...
0038d0	38 00 00 00 00 03 00 17 00 00 00 00 00 01 04 00	8.....
0038e0	01 07 06 01 07 06 02 07 01 02 02 04 01 04 08 01
0038f0	01 09 02 02 06 01 02 01 1B 43 44 4E CDN

The status bar at the bottom indicates "3 byte(s) selected" and "Pos: 38F9h (14585)".

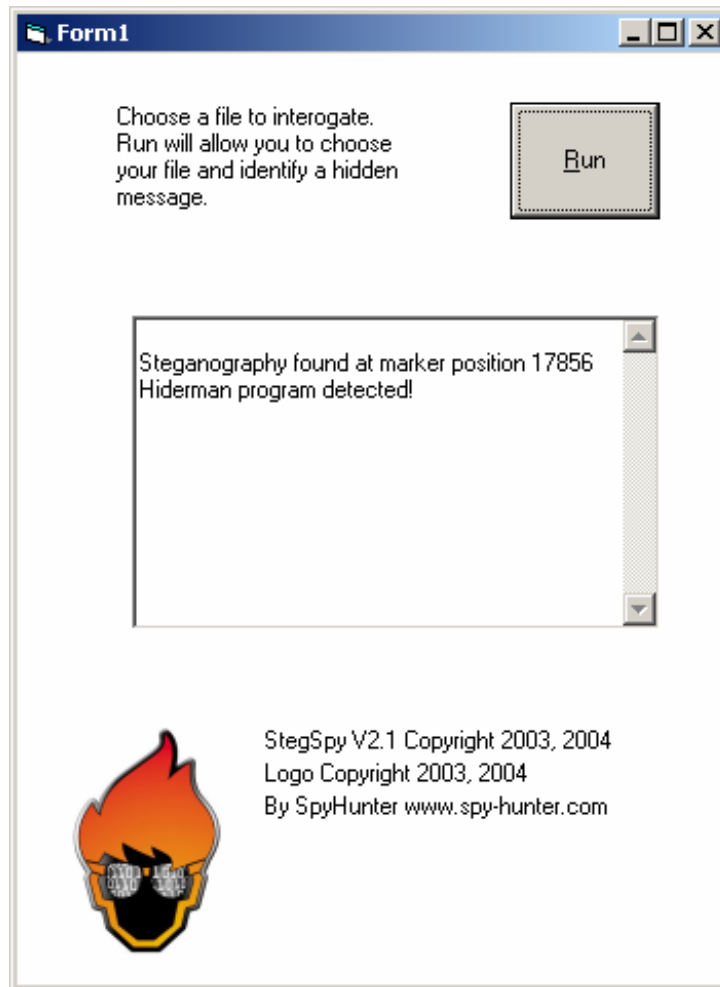
Steganalysis – Stegspy V2.1

- ▶ **StegSpy V2.1**
 - **Signature identification program**
 - **Searches for stego signatures and determines the program used to hide the message**
 - **Identifies 13 different steganography programs**
 - **Identifies location of hidden message**



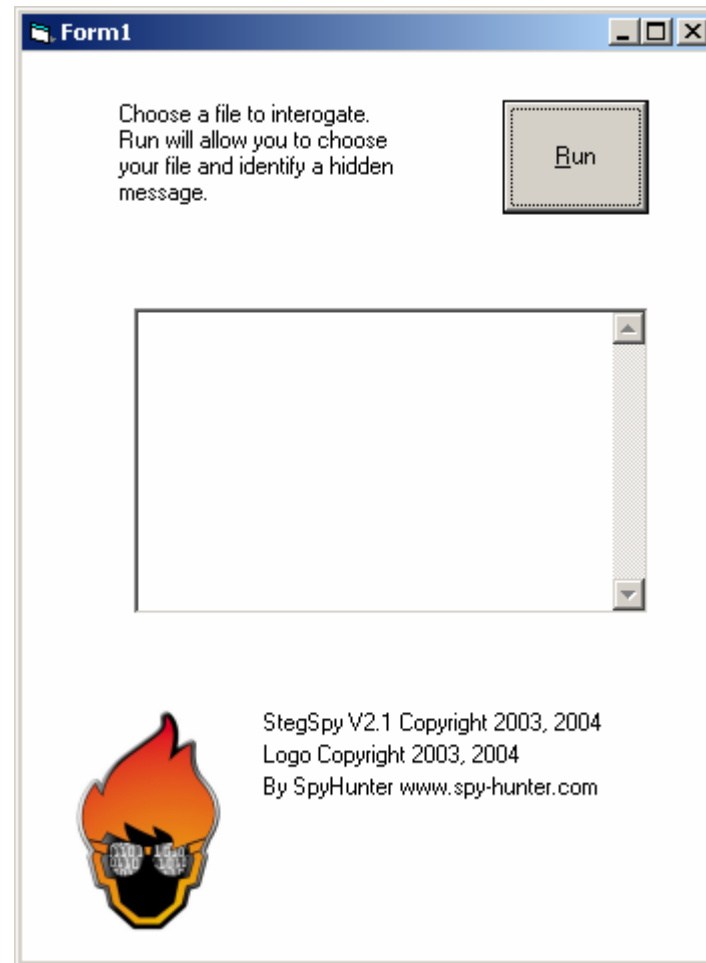
Steganalysis - Stegspy

► StegSpy - Demo



Steganalysis – Stegspy V2.1

- ▶ **StegSpy V2.1**
 - Available for download from my site
 - ▶ www.spy-hunter.com
 - Features currently under development:
 - ▶ New signatures
 - ▶ Scanning entire directories or drive
 - ▶ A *NIX-friendly version of the program



Steganalysis – Identifying a signature

- ▶ **Signature-based steganalysis was used to identify signatures in many programs including Invisible Secrets, JPHide, Hiderman, etc.**



Steganalysis – Identifying a signature

- ▶ **How is this handy?**
- ▶ **No original file to compare it to**
- ▶ **Search for the signature pattern to determine the presence of a hidden message**
- ▶ **Signature reveals program used to hide the message!**



Steganalysis meets Cryptanalysis

Revealing hidden files

Steganalysis meets Cryptanalysis

Cryptanalysis

- ▶ **As stated previously, in Steganography the goal is to hide the message, NOT encrypt it**
- ▶ **Cryptography provides the means to encrypt the message.**
- ▶ **How do we reveal the hidden message?**



Steganalysis meets Cryptanalysis

- ▶ **Knowing the steganography program used to hide the message can be extremely handy when attempting to reveal the actual hidden message**
- ▶ **Identifying and cracking the algorithm**
 - Unfortunately, some of these programs use strong encryption 256-bit or stronger – GOOD LUCK!
- ▶ **Reveal or Crack the password, seed, or secret key**
 - Practically all Steganography programs use a password to hide the message



Cryptanalysis

- ▶ **Identify program used to hide message**
- ▶ **Identify the location of the program signature in the file**
- ▶ **Identify the location of the password in the file**
- ▶ **Identify location of the hidden message in the file**
- ▶ **Identify the algorithm used to encrypt the hidden message**



Steganalysis – Password Guessing

Password Guessing/Dictionary Attacks

- ▶ **A few password guessing programs have been created.**
- ▶ **Stegbreak by Niels Provos, www.outguess.org**
 - J-Steg
- ▶ **Can now be found on the Knoppix Penguin Sleuth forensics CD**
 - www.linux-forensics.com



Cryptanalysis – Brute Force Method

Brute Force – Reverse Engineering

▶ Common encryption techniques

- Modification of LSB (Least Significant Bit)
- Password and/or contents masked using an algorithm
 - ▶ Algorithm based on a secret key
 - ▶ Algorithm based on the password
 - ▶ Algorithm based on a random seed hidden somewhere else in the file

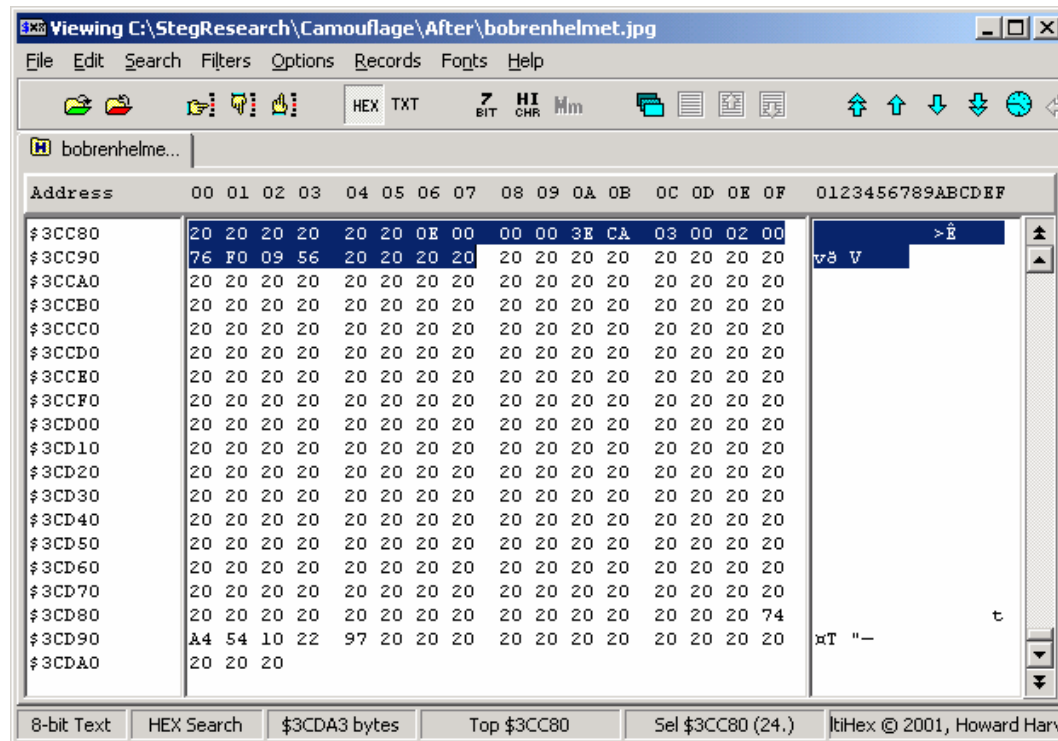


Cryptanalysis – Brute Force Method

- ▶ **Common encryption algorithms used in steganography programs**
 - XOR
 - DES
 - 3DES
 - IDEA
 - AES

Camouflage – Case Study

- ▶ **Determining the password used with Camouflage**
- ▶ **The location of the password was determined by using MultiHex which allows searches for Hex strings**



Camouflage

- ▶ **The string was found to be "76 F0 09 56"**
- ▶ **The password is known to be "test" which is "74 65 73 74" in Hex**



BDHTool

- ▶ **BDHTool we can XOR the two to reveal the key**

The screenshot shows the 'LOGIC OPERATORS V1.2' application window. It features two input sections for 8-bit bytes, A and B, and a central logic operator panel. The 'XOR' operator is selected and highlighted in red. Below the operator panel is a truth table for XOR and a logic symbol. To the right is an 'RPN Calculator' with a numeric keypad and function buttons. At the bottom are 'CLEAR' buttons for each section.

HEX	55	NOT	HEX	74	NOT	DEC	34	HEX	22
BIN	01010110		BIN	01110100		BIN	00100010		
Bit Select	0 0 0 0 0 0 0 0		Bit Select	0 0 0 0 0 0 0 0					
Low Nibble	x0 x1 x2 x3 x4 x5 x6 x7		Low Nibble	x0 x1 x2 x3 x4 x5 x6 x7					
Hi Nibble	0x 1x 2x 3x 4x 5x 6x 7x		Hi Nibble	0x 1x 2x 3x 4x 5x 6x 7x					

AND
NAND
OR
NOR
XOR
XNOR
NOT

XOR

A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

HELP INFO
CLEAR BYTE A
CLEAR BYTE B
CLEAR RESULT
CLEAR ALL

Camouflage

76 XOR 74 = 02

F0 XOR 65 = 95

09 XOR 73 = 7A

56 XOR 74 = 22

- ▶ **The 1st 4 digits of the key are "02 95 7A 22"**
- ▶ **So let's test our theory...**

Camouflage

- ▶ **We store another message using a different password**
- ▶ **The file reveals a Hex code of "63 F4 1B 43"**
- ▶ **We XOR this with the known key "02 95 7A 22"**
- ▶ **The result is "61 61 61 61" which is a password of "aaaa" in ASCII**
- ▶ **We've revealed the hidden password to hide the message!**
- ▶ **This exploit discovered by Guillermito at www.guillermito2.net**



Forensics/Anti-Forensics

Anti-Forensics

- ▶ **Best Practices when using Steganography programs:**
 - Use a password different than your O/S password
 - Delete original message once you have created a new image with the hidden message
 - Remove the Steganography program after hiding the message
 - OR run the Steganography program from a CD if possible.
 - Use Alternate Data Streams...



Anti-Forensics – Alternate Data Streams

▶ Alternate Data Streams

- (NTFS) New Technology File System allows for Alternate Data Streams
- One file can be a link to multiple Alternate Data Streams of files of any size.
- Important Note! – These Alternate Data Streams are Hidden!
- Allows for hiding of files and even directories!
- Difficult to detect
 - ▶ Doesn't show up when you run `c:\dir`

Anti-Forensics – Alternate Data Streams

▶ Alternate Data Streams

- C:\notepad mike.txt:mikehidden.txt
- This allows mikehidden.txt to be a hidden ADS
- C:\dir
02/26/2004 02:29p 0 mike.txt
- Notice – no indication of mikehidden.txt
- Although a message was saved in the mikehidden.txt, the mike.txt shows 0 bytes!

Windows – Alternate Data Streams

▶ Alternate Data Streams

- Lets save a 2nd file
- C:\notepad mike.txt:mikehidden2.txt
- This allows mikehidden2.txt to be a hidden ADS
- C:\dir
02/26/2004 02:29p 0 mike.txt
- Still no indication of mikehidden.txt or mikehidden2.txt

Anti-Forensics – Alternate Data Streams

- ▶ Alternate Data Streams can be used to hide private files, viruses, trojans, and steganography!
 - **Anti-Virus/Anti-Trojan Test - Does your scanner pass the test?**
 - There's a small utility MakeStream, that can be used to move a virus or trojan to a hidden Alternate Data Stream attached to an "innocent" text file!
 - For example, if you ran **makestrm.exe c:\test.exe**, the file contents of c:\test.exe would be moved into c:\test.exe:StreamTest (an Alternate Data Stream), and the original file contents are then over-written with a simple message reminding you about the linked stream.
 - Get any trojan or virus that is detected by your virus/trojan scanner, and run makestrm.exe on it to move its file contents into a hidden stream. Then, re-scan the file - is it still detected?
 - Many commercial scanners **do not** identify viruses and trojans hidden in ADS's!
 - <http://www.diamondcs.com.au/web/streams/streams.htm>



Forensics

- ▶ **If performing Forensics and discover a potentially “stega-nized” file:**
 - Look for evidence of steganography programs on the computer
 - Leverage other O/S and application passwords found on the machine, this may also be the password used to hide the message
 - Look for other hints such as a password written down on a note, letters, diaries, etc.
 - For more info – please see “Electronic Crime Scene Investigation – A Guide for First Responders, U.S. Dept of Justice”



Forensics – Alternate Data Streams

► Tools for Detecting Alternate Data Streams

- LNS – www.ntsecurity.nu
- LADS - www.heysoft.de
- NTFS ADS Check - www.diamondcs.com.au

```
C:\WINNT\System32\cmd.exe
C:\Downloads\LNS>ls c:\tools\ads

ls 1.0 - (c) 2002, Arne Vidstrom (arne.vidstrom@ntsecurity.nu)
- http://ntsecurity.nu/toolbox/ls/

c:\tools\ads\mike.txt
- Alternative data stream [:mikehidden.txt:$DATA]
c:\tools\ads\mike.txt
- Alternative data stream [:mikehidden2.txt:$DATA]
C:\Downloads\LNS>
```



Conclusions

Steganalysis – Future?

- ▶ **Where do we go from here?**
- ▶ **My program StegSpy currently identifies JPHide, Hiderman, and Invisible Secrets. More to come!**
- ▶ **Write a program to crack weak Stego programs**
- ▶ **Need a password grinder, may vary depending on the Stego program (stegbreak already available)**
- ▶ **Statistical analysis has been performed and is also capable of detecting Steganographic programs (histogram, LSB, etc)**

Steganalysis – Other Tools

- ▶ **Wetstone Technologies offers Stego Watch**
- ▶ **Identifies the presence of steganography through special statistical and analytical programs.**
- ▶ **Accurate and comprehensive tool (\$\$\$)**
- ▶ **Does not attempt to crack or reveal the hidden message, merely identifies it**
- ▶ **Offer a Steganography Investigator Training Course**
- ▶ **See <http://www.wetstonetech.com>**



Steganalysis – Other Tools

- ▶ **Stegdetect by Niels Provos**
- ▶ **Available at** <http://www.outguess.org/detection.php>
- ▶ **Detects**
 - jsteg
 - jphide (unix and windows)
 - invisible secrets
 - outguess 01.3b
 - F5 (header analysis)
 - appendX and camouflage
- ▶ **Site down due to State of Michigan law!**



References

- ▶ **Steganographica, Gaspari Schotti, 1665**
- ▶ **Disappearing Cryptography, Peter Wayner, 2002**
- ▶ **Hiding in Plain Sight, Eric Cole 2003**
- ▶ **Steganography – presentation Chet Hosmer, Wetstone Technologies, TechnoSecurity 2003**





Question and Answer