

Steganographic Techniques on Social Media: Investigation Guidelines

AIMIE CHEE

B.S. Computer Science (Southwest Minnesota State University, USA)

A thesis submitted to the graduate Faculty of Design and Creative Technologies
AUT University
in partial fulfilment of the
requirements for the degree of
Master of Forensic Information Technology

School of Computing and Mathematical Sciences

Auckland, New Zealand
2013

Declaration

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which, to a substantial extent, has been accepted for the qualification of any other degree or diploma of a University or other institution of higher learning, except where due acknowledgement is made in the acknowledgements.

.....

Aimie Chee

(20 May 2013)

Acknowledgements

This thesis was completed at the Faculty of Design and Creative Technologies in the School of Computing and Mathematical Sciences at Auckland University of Technology. I would like to express my deepest gratitude to everyone who has supported me through the two years of my thesis journey. First of all, I would like to thank my father, Shout Twong, and my mother, You Laa, who have financially supported my entire post graduate study and have given me the courage to fulfil my dream. I would also like to thank my best friend, Cathy, who has continuously guided, helped, and encouraged me whenever I was stressed and lost during my post graduate study.

I would like to thank my thesis supervisor, Dr. Brian Cusack, who has provided valuable advice on and inspiration for the thesis project, without him, I would not have been able to achieve this much. Thanks also to my course mates, Ting and Muteb for sharing their brilliant ideas and providing valuable suggestions in regard to my thesis project. I would like to offer my deepest appreciation to all the postgraduate staff and lecturers for supporting my two years of postgraduate study, without you, I would not have successfully completed my degree. I appreciate the services of Catriona Carruthers who proof-read this dissertations.

Last but not least, I would like to thank Magnet Forensic software for providing the full trial version of Internet Evidence Finder and Mr. Jad Saliba from the Magnet Forensic support desk for supporting me and addressing the Facebook chat recovery issues that occurred during the execution of this thesis project. I would also like to extend my appreciation to Backbone Security for providing me with the opportunity of using their trial version of StegAlyzerAS and StegAlyzerSS and to thank Mr. Robert W. Lipscomb for answering queries in regard to the tool's technical issues.

Abstract

Online social networking is available to anyone who wants to sign up to the many sites available. The web-based services allow users to communicate with many media sources and to build relationship networks that have personal meaning. The medium permits open communication and, consequently, the propagation of hidden messages (steganography) and the exchange of images, text, sound files and so on, that may contain hidden information. The purpose of this research is to find out whether or not it is necessary to include steganography as a routine check when conducting digital forensics examinations in relation to online social networking. This is a challenge to digital forensic investigators as the hidden messages will not be found if they are not being searched for.

The research testing was carried out in a laboratory environment under an empirical approach. In the pre-test, five steganographic techniques with different image formats were uploaded on Facebook and Google+ social network websites and then downloaded to identify the techniques that can and cannot be used on Facebook and Google+ for the complete process of covert communication up to the extraction of the hidden messages. Two suitable techniques, JP Hide and Seek and StegHide with common JPEG images were chosen for the experimental case scenarios, based on the pre-test results. The experimental case scenarios were simulated on laboratory computers and digital forensic examinations were undertaken to identify both the uploaded hidden messages in different images and to extract the hidden messages in the uploaded and downloaded image files. Based on the digital forensic examination performed on the experimental case scenarios, a guideline for the steganographic examination process was established.

The findings from the pre-test results showed that steganography is difficult to perform in the Facebook photo upload feature. Here the hidden message cannot be extracted after the image is downloaded from Facebook, but it can be successfully performed through the message file attachment and group file sharing features with a variety of image formats such as JPEG, PNG, BMP, and GIF. On Google+ photo sharing, on the other hand, the complete cycle of steganography communication from embedding up to the extraction of hidden messages was successfully undertaken with JPEG, PNG, BMP or GIF image

formats. The results show that steganography can be propagated in social media; therefore it is necessary to include steganographic evaluation in the standard digital investigation procedures.

It was discovered during the research experiment that there is a lack of effective forensic tools in the area of steganographic image analysis or signature detection. The current steganalysis tools are designed for specific signatures but there are very many steganographic tools that are capable of embedding hidden messages using different techniques. This is a challenge for the digital forensic investigator. Therefore, there is an opportunity for further research in this area where the capabilities of detection tools can be further developed with more steganographic signatures.

Table of Contents

Declaration.....	ii
Acknowledgements	iii
Abstract.....	iv
Table of Contents	vi
List of Tables	xii
List of Figures	xiv
List of Abbreviations.....	xvi

Chapter 1: Introduction

1.0 BACKGROUND.....	1
1.1 MOTIVATIONS	2
1.2 THE RESEARCH APPROACH.....	4
1.3 THE RESEARCH FINDINGS	4
1.4 STRUCTURE OF THESIS	6

Chapter 2: Literature Review

2.0 INTRODUCTION.....	8
2.1 STEGANOGRAPHY OVERVIEW	8
2.1.1 Steganography vs Cryptography	10
2.1.2 Steganography vs Watermarking	11
2.1.3 The Prisoner's Problem	12
2.1.4 Steganography Classification.....	13
2.1.4.1 Semagrams	14
2.1.4.2 Open Codes	14
2.1.4.3 Spam Mimics.....	15
2.1.4.4 Digital Media.....	16
2.1.4.5 Disk Space.....	17
2.1.4.6 Protocol.....	17
2.1.4.7 Other Files.....	17
2.2 DIGITAL IMAGE FORMAT.....	18
2.2.1 Colour Representation	19

2.2.2	Raster Format	20
2.2.3	Palette Format	21
2.2.4	Transform Format - JPEG.....	21
2.3	IMAGE STEGANOGRAPHY	22
2.3.1	Text File (.txt) Injection into Image File	23
2.3.2	Zip File (.rar / .zip) Injection into Image File	24
2.3.3	Hiding in EXIF.....	26
2.3.4	Least Significant Bits (LSB) Substitution in Spatial Domain Images	27
2.3.5	Least Significant Bits (LSB) Substitution in DCT	30
2.4	SOCIAL NETWORK PHOTO SHARING CAPABILITIES	32
2.4.1	Facebook Photo Sharing	34
2.4.2	Google+ Photo Sharing	35
2.5	DIGITAL FORENSICS	36
2.5.1	Social Network Forensics	38
2.5.2	Web Forensics	41
2.5.3	Steganalysis.....	44
2.6	SUMMARY OF ISSUES AND PROBLEMS.....	46
2.7	CONCLUSION	48

Chapter 3: Research Methodology

3.0	INTRODUCTION.....	50
3.1	REVIEW OF SIMILAR RESEARCH	51
3.1.1	Searching for Hidden Messages	51
3.1.2	Detecting Steganographic Content on the Internet	53
3.1.3	Forensic Artefacts of Uninstalled Steganography Tools	56
3.1.4	Effective Digital Forensic Analysis of the NTFS Disk Images	58
3.1.5	Computer Forensic Guidance Model with Case Study	60
3.2	RESEARCH DESIGN.....	63
3.2.1	Summary of Similar Studies	64
3.2.2	Review of Problem Areas	66
3.2.3	The Research Question and Hypotheses.....	67
3.2.4	Research Phases	68
3.2.5	Data Map.....	70

3.3	DATA REQUIREMENTS	71
3.3.1	Investigation Case Scenarios	71
3.3.1.1	Terrorism – Case Scenario 1	71
3.3.1.2	Intellectual Property – Case Scenario 2.....	72
3.3.2	Data Collection.....	73
3.3.3	Data Processing	75
3.3.4	Data Analysis	76
3.3.5	Data Presentations	77
3.4	LIMITATIONS	78
3.3	CONCLUSION	80

Chapter 4: Research Findings

4.0	INTRODUCTION.....	81
4.1	VARIATION ENCOUNTERED IN EXPERIMENT.....	81
4.1.1	Case Scenario	81
4.1.2	Data Collection.....	82
4.1.3	Data Processing	83
4.1.4	Data Analysis and Presentation.....	84
4.2	SOCIAL MEDIA PRETEST	84
4.2.1	Environment Set Up	85
4.2.2	Findings	86
4.2.3	Social Media: Steganographic Techniques	91
4.2.3.1	Facebook	91
4.2.3.2	Google+.....	91
4.2.4	Conclusion	93
4.3	CASE SCENARIO 1 - TERRORISM	94
4.3.1	Environment Set Up	95
4.3.2	Digital Forensics	95
4.3.2.1	Evaluation and Assessment	96
4.3.2.2	Acquisition of Digital Evidences.....	96
4.3.2.3	Survey the Digital Scene	97
4.3.2.4	Digital Evidence Examination.....	97

4.3.2.5	Reconstruction of Extracted Data	100
4.3.2.6	Conclusion.....	109
4.3.3	Comparative Analysis.....	110
4.4	CASE SCENARIO 2 - INTELLECTUALPROPERTY	111
4.4.1	Environment Set Up	112
4.4.2	Digital Forensics	112
4.4.2.1	Evaluation and Assessment	112
4.4.2.2	Acquisition of Digital Evidences.....	112
4.4.2.3	Survey the Digital Scene	113
4.4.2.4	Digital Evidence Examination.....	113
4.4.2.5	Reconstruction of Extracted Data.....	115
4.4.2.6	Conclusion.....	120
4.4.3	Comparative Analysis.....	120
4.5	CONCLUSION	122

Chapter 5: Research Discussion

5.0	INTRODUCTION.....	124
5.1	ANSWERING THE RESEARCH QUESTION	124
5.1.1	Answers to Sub-Questions.....	125
5.1.2	Hypothesis Tests	131
5.1.3	The Research Question Answer	137
5.2	DISCUSSION	139
5.2.1	Discussion on Case Scenario Environment	140
5.2.2	Discussion on Data Acquisition and Extraction.....	141
5.2.3	Discussion on Reconstruction & Analysis.....	142
5.2.4	Recommendation on Steganography Evaluation	144
5.3	CONCLUSION	145

Chapter 6: Conclusion

6.0	CONCLUSION	146
6.1	LIMITATIONS OF RESEARCH	148
6.2	FUTURE RESEARCH.....	150

REFERENCES	153
------------------	-----

APPENDICES	165
Appendix 1 Possible Errors from StegDetect	165
Appendix 2 Scenario 1 Experimental Images before JP Hide and Seek Steganographic Process (1 False Positive)	165
Appendix 3 Scenario 1: Christian Riley's Imaged Hard Drive Verification	166
Appendix 4 Scenario 1: John Doe's Imaged Hard Drive Verification..	167
Appendix 5 Scenario 1: Christian Riley's Imaged RAM Verification..	168
Appendix 6 Scenario 1: John Doe's Imaged RAM Verification	169
Appendix 7 Scenario 1: John Doe's Imaged Hard Drive Verification..	169
Appendix 8 Facebook Pre-Test Photo Identifier	171
Appendix 9 Google+ Pre-Test Photo Identifier	171
Appendix 10 Facebook Pre-Test Configuration and Results.....	172
Appendix 11 Google+ Pre-Test Configuration and Results	180
Appendix 12 Scenario 1 Simulation Control Data (Target Machine 1 - Christian Riley)	183
Appendix 13 Scenario 1 Simulation Control Data (Target Machine 2 – John Doe).....	186
Appendix 14 Scenario 2 Simulation Control Data (Target Machine 3 – John Doe).....	190
Appendix 15 Scenario 1 JP Hide and Seek's artefacts detected by StegAlyzerAS (Target Machine 1 – Christian Riley).....	193
Appendix 16 Scenario 1 JP Hide and Seek's artefacts detected by StegAlyzerAS (Target Machine 2 –John Doe)	193
Appendix 17 Scenario 1 Bon Kyu Bon's artefacts detected by StegAlyzerAS (Target Machines 1 & 2 – False Positive) .	194
Appendix 18 Scenario 1 Facebook File Download Artefacts (Target Machine 1 – Christian Riley)	195
Appendix 19 Scenario 1 Facebook File Download Artefacts (Target Machine 2 – John Doe).....	197
Appendix 20 Scenario 1 Facebook File Upload URL History (Target Machine 1 – Christian Riley)	199
Appendix 21 Scenario 1 Facebook File Upload URL History (Target Machine 2 –John Doe).....	201

Appendix 22 Scenario 1 Facebook File Upload Artefacts (Target Machine 1 – Christian Riley)	204
Appendix 23 Scenario 1 Facebook File Upload Artefacts (Target Machine 2 – John Doe)	212
Appendix 24 Scenario 1 Images of Interest in Suspects' Hard Drives ...	219
Appendix 25 Scenario 1 Facebook Chat Artefacts from pagefile.sys and unallocated cluster	224
Appendix 26 Scenario 2 Suspect Google+ Account Artefact.....	234
Appendix 27 Scenario 2 Google+ Photo Upload URL History	235
Appendix 28 Scenario 2 Suspected Images found in Browser Cache.....	236
Appendix 29 Scenario 2 Suspected Steganographic Images in Suspect's Hard Drive.....	243
Appendix 30 Scenario 2 Suspicious File Activities	244
Appendix 31 Scenario 2 Google+ Message Posted (Keyword Search) ..	253
Appendix 32 Scenario 2 Significant Registry Artefacts Identified by StegAlyzerAS on a portable StegHide application	255

List of Tables

Table 2.1: Information Hiding Using HTML	18
Table 2.2: LSB Substitution Table	28
Table 2.3: OSN Pre-processing activities on uploaded images	33
Table 2.4: Browser Cache and Internet History File Locations for Internet Explorer, Firefox, and Google Chrome	43
Table 3.1: Percentages of false positives from Images obtained from the Internet	55
Table 4.1: Facebook Photo Upload Results.....	87
Table 4.2: Google+ Photo Upload Results	88
Table 4.3: Facebook File Sharing Results	89
Table 4.4: Facebook Message Attachment Results.....	90
Table 4.5: Steganographic techniques supported or inhibited on Facebook	92
Table 4.6: Steganographic techniques supported in Google+	93
Table 4.7: StegAlyzerAS and StegAlyzerSS Detection Summary (Senario 1).....	97
Table 4.8: Summary of Facebook Related Internet Activities.....	98
Table 4.9: Reconstructed Facebook Chat	101
Table 4.10: Identified Image File Locations.....	105
Table 4.11: Timeline Analysis (Secnario 1).....	108
Table 4.12: Scenario 1 Comparative Analysis.....	110
Table 4.13: StegAlyzerAS and StegAlyzerSS Detection Summary (Scenario 2)	113
Table 4.14: Google+ Internet History Data Extracted	114
Table 4.15: Scenario 2 Reconstructed Message Posted	119
Table 4.16: Scenario 2 Comparative Analysis.....	121
Table 5.1: Sub-Question 1 and Answer.....	125
Table 5.2: Sub-Question 2 and Answer.....	126
Table 5.3: Sub-Question 3 and Answer.....	127
Table 5.4: Sub-Question 4 and Answer.....	128
Table 5.5: Sub-Question 5 and Answer.....	128
Table 5.6: Sub-Question 6 and Answer.....	129
Table 5.7: Sub-Question 7 and Answer.....	130

Table 5.8: Sub-Question 8 and Answer.....	131
Table 5.9: Sub-Question 9 and Answer.....	131
Table 5.10: Tested Hypothesis 1.....	132
Table 5.11: Tested Hypothesis 2.....	134
Table 5.12: Tested Hypothesis 3.....	136
Table 5.13: Research Main Question and Tested Hypothesis	137

List of Figures

Figure 1.1: Suggested Steganographic Evaluation Flow Chart	5
Figure 2.1: Steganographic System Mechanism.....	10
Figure 2.2: Steganographic Taxonomy	13
Figure 2.3: An example of a spam mimic message generated using the web-based tool provided by www.spammimic.com	16
Figure 2.4: RGB Colour Intensity Representations	19
Figure 2.5: Cover-object.....	23
Figure 2.6: Stego-object	23
Figure 2.7: Secret message revealed through Notepad application	24
Figure 2.8: Stego-object produced by appending the text file (.txt) into the image file.....	25
Figure 2.9: Stego-object produced by appending a zip file (.rar) into an image file.....	26
Figure 2.10: Secret data extracted using the WinRAR application	26
Figure 2.11: A secret recipe was hidden in an image's EXIF properties.....	27
Figure 2.12: BMP format cover-object with an original size of 663kb	29
Figure 2.13: BMP stego-object embedded with 1.04kb of secret message created using the Hide in Picture steganography tool.....	30
Figure 2.14: JPEG format cover-object with an original size of 120kb.....	31
Figure 2.15: JPEG stego-object embedded with 1.04kb of secret message created using the JP Hide and Seek steganography tool	32
Figure 2.16: StegDetect developed by Neils Provos.....	46
Figure 3.1: Forensic investigation steps	59
Figure 3.2: Model Phases	61
Figure 3.3: Digital Forensic Phase	61
Figure 3.4: Digital Evidence Acquisition.....	63
Figure 3.5: Research Phases	69
Figure 3.6: Proposed Research Data Map	70
Figure 4.1: Facebook Home Page Layout	85
Figure 4.2: Google+ Home Page Layout.....	85
Figure 4.3: Lab environment steganography process.....	86

Figure 4.4: Facebook default picture file name when download.....	88
Figure 4.5: Facebook group file sharing feature.....	89
Figure 4.6: All the files that the user shared with the Melody group.....	90
Figure 4.7: Steganographic images can be attached in the message by using the Add Files function.....	91
Figure 4.8: Steganographic image generated by JP Hide & Seek (left) and Steganographic image generated by SilentEye (right).....	94
Figure 4.9: Data Acquisition Process.....	95
Figure 4.10: Prefetch file extraction on Christian Riley's machine.....	99
Figure 4.11: Prefetch file extraction on John Doe's machine	99
Figure 4.12: Keyword search extracted from Christian Riley's imaged hard drive.....	100
Figure 4.13: Facebook page fragment artefacts for file upload.....	103
Figure 4.14: Identified image file with appended data.....	106
Figure 4.15: Steganographic images detected by StegDetect on Christian Riley's imaged hard drive.....	107
Figure 4.16: Steganographic images detected by StegDetect on John Doe's imaged hard drive.....	108
Figure 4.17: .txt files that extracted by Encase (John Doe).....	109
Figure 4.18: Most active STEGHIDE.EXE prefetch file that found on John Doe's machine.....	114
Figure 4.19: Files contained in STEGHIDE.EDE-0AB8EA11.pf.....	115
Figure 4.20: HEX value in the header of the suspect steganographic image	116
Figure 4.21: HEX value in the header of a regular, clean digital image	116
Figure 4.22: Displayed images of interest found in the web browser cache files by EnCase	118
Figure 4.23: Suspicious activities timeline.....	119

List of Abbreviations

• APP	Application
• ARP	Address Resolution Protocol
• ASCII	American Standard Code for Information Interchange
• BMP	Bitmap Image File
• BOF	Beginning of File
• DCO	Device Configuration Overlays
• DCT	Discrete Cosine Transform
• dd	Disk Dump
• DFRWS	Digital Forensic Research Workshop
• DOS	Disk Operating System
• DWT	Discrete Wavelet Transform
• EOF	End of File
• EXIF	Extended File Information
• GIF	Graphics Interchange Format
• HDD	Hard Disk Drive
• HEX	Hexadecimal
• HPA	Host Protected Areas
• HTML	Hyper Text Markup Language
• HVS	Human Visual System
• ICMP	Internet Control Message Protocol
• ID	Identity
• IE	Internet Explorer
• IPR	Intellectual Property Rights
• JPEG	Joint Photographic Experts Group
• KB	Kilobytes
• LAN	Local Area Network
• LSB	Least Significant Bits
• MD5	Message Digest
• MFT	Master File Table
• ML	Machine Learning

- MSB Most Significant Bit
- MSN Microsoft Network
- NIST Nation Institute of Standards and Technology
- NTFS New Technology File System
- OSN Online Social Networking
- PNG Portable Network Graphics
- RAM Random Access Memory
- RGB Red, Green, Blue
- SARC Steganography Analysis and Research Center
- TCP/IP Transmission Control Protocol/Internet Protocol
- TIFF Tagged Image File Format
- UDP User Datagram Protocol
- URL Uniform Resource Locator
- VM Virtual Machine
- XML Extensible Markup Language
- $YCrCb$ Linear transformation of blue-luminance (U) and red-luminance (V) colour model
- YUV Luminance (Y), blue-luminance (U), red-luminance (V)

Chapter 1

Introduction

1.0 BACKGROUND

Steganography can be defined as “the art of hiding information in ways that prevent the detection of a hidden message” (Johnson & Jajodia, 1998, p.26). Steganography has been used since ancient times, when more physical approaches were employed, such as the use of invisible ink, wax, microdots, and tattooing on the scalps of slaves (Fridrich, 2010). Today, steganography techniques have been digitalized with the advent of the personal computer and other advances in technology. Today’s steganographic tools can hide any type of binary data in nearly any type of multimedia or data file (Kessler, 2004a). From a visual perspective, steganography is preferable to cryptography because of its innocent appearance (known as the cover-object) which may mean adversaries may not even notice the existence of a secondary message channel. In contrast, the scrambled text of cryptography may itself draw the attention of adversaries to detect, intercept and modify the messages (Dunbar, 2002; Engle, 2003).

The emergence of online social networking (OSN) has encouraged a new channel of communication where people can send messages, share their photos, videos, and information, becoming pervasive in daily life. There has been a massive increase in the use of OSN, which facilitates a high degree of user intercommunication (Zainudin, Merabti, & Llewellyn-Jones, 2010). When use is widespread, there is a higher chance of misuse. Acohido (2011) from ‘*USA Today*’ reported that sex predators are now targeting children via online social media. Mostyn (2010) also indicated that Facebook, one of the leading OSN websites, is becoming the repository of crimes, according to the United Kingdom police, ranging from fraud, acts of terrorism, illegal firearm, trafficking to harassment.

Technologically-minded criminals illegally used technology for profit in the same way business people use it legally. Criminals use of technology is growing in ways previously unanticipated (Castiglione, D’Alessio, & De Santis, 2011) and criminals are now becoming more sophisticated and rigorous in their

attempts to use technology in order to evade detection and facilitate their crimes (Zainudin, Merabti, & Llewellyn-Jones, 2011). Thus, the utilization of steganography in OSN should be anticipated. For example, text and images are common artefacts in OSN. Perpetrators may send steganographic text or images by posting them on their OSN so that only the intended receiver can download the steganographic objects and retrieve the hidden message. By using this method, covert communication is not obvious, as these steganographic texts or images appear as ordinary user generated content (Castiglione, D'Alessio, et al., 2011). On the other hand, it is also possible to use steganography to store legally information on an OSN and retrieve it as needed (Kumar & Pooja, 2010). It is, therefore, very useful to identify how an old trick – steganography, operates in a new context.

OSNs can offer new opportunities and the digital forensic investigator has to be alert when such a situation occurs. Steganography creates a new challenge to the digital forensic investigator. The reason why it is not always used is due to the fact that most investigators do not routinely search for steganographic tools and frequently use immature methods when looking for steganographic content (Kessler, 2004a). Accordingly, the research question in regard to the topic is stated:

Should digital forensic investigators include a routine steganography check as part of their standard procedure during a digital forensic investigation in relation to online social networks?

1.1 MOTIVATION

A keyword search for steganography through the internet was performed in 1996 with a list of less than twelve hits. In 1998, it had over a thousand hits. In 2008 it returned 2.2 million hits from a Google search (Curran & Devitt, 2008). Obviously, the research in this area has rapidly developed over time. However, an average criminal may not know what steganography is, but they may understand hiding information or information hiding. Surprisingly, a Google search for this word returned an astonishing number of hits when this research was conducted. It returned over 80 million references! To imagine ways that a criminal may utilize steganography to hide information is easy, but do law enforcement agents include steganography searches as part of their routine checks when they are conducting

an investigation? What procedural practices do they have? What tools are used? According to the US National Institute of Justice, the most common illegal use of steganography is for the possession and storage of child pornography images. It may well be used to commit fraud, terrorist activities and other illegal acts also (National Institute of Justice, 2010).

Recently, law enforcement agencies have reported using OSN as a tool in their investigations and gaining evidence from wall posts, messages, and photos (Hayes, 2011; Ruotolo, 2012; Scoville, 2011). According to the U.S. Citizenship and Immigration Services (USCIS), more than 62% of agencies polled across 48 states and the District of Columbia acknowledged using social media searches in criminal investigations (Scoville, 2011). What would happen if steganography had been used on OSN as suggested in the proposed research? Would the viable digital evidence still have been found? It is, therefore, worthwhile looking at how to perform digital forensics examinations to obtain and preserve the integrity of probative digital evidence in relation to activities employing steganography in OSNs. Probative evidence may not be seen or found if it is not being looking for (Kessler, 2004a). So far there has been no significant study conducted from the digital forensic perspective on this meeting of steganography and online social networking. This is a growing area of research as steganography techniques will only become more sophisticated and more beneficial to criminals. It will be challenging to the investigator, if the appropriate tools and techniques to investigate this area do not improve in line with the development of steganography. The popularity of online social networking is still at its peak and crimes that are involved with social networking have risen eight-fold since 2008, as reported by the UK BBC news ("Huge rise in social media 'crimes'," 2012). In New Zealand also, it has been reported that social media is a target for criminals and a hotbed for cyber criminal activity (Chapman, 2011; Wade, 2012). Therefore, it is critical for an investigator to be very familiar with the procedures and practices before the problem arises.

1.2 THE RESEARCH APPROACH

In order to answer the research question, research is to be conducted from an empirical approach. The selected approach was chosen from a review of five previous research reports from similar areas to ensure that the proposed research is conducted with an effective research methodology. Nine research sub-questions that are relevant to the research experiment were constructed. Sub-questions were also developed to aid the testing of the three asserted hypotheses.

In line with the research sub-questions and the asserted hypotheses, the research is divided into five phases. In Phase 1, research was conducted with a pre-test to identify possible ways of using steganography in social media. Then, based on collected pre-test results, two steganographic techniques were selected and implemented in simulated case scenarios. Phase 3 and Phase 4 were designed to discover an effective way of conducting steganographic evaluation in a forensically sound manner on the simulated case scenarios. That learned and observed in Phases 3 and 4 were then reflected in Phase 5 as a recommendation for processes or procedures for steganographic evaluation.

1.3 THE RESEARCH FINDINGS

The research has proved that the application of steganography in social media is possible. When the incriminating data is hidden and disseminated in this way, and steganographic evaluation is not included as a standard digital forensic examination procedure, the incriminating data can pass without notice. Therefore, it is necessary to include steganographic evaluation as a standard procedure when conducting digital forensic investigation. The experimental research found that image steganography using JP Hide and Seek, Silent Eye, End of File (EOF) append, StegHide, S-tools, and Invisible Secret 4 techniques are unlikely to be propagated in Facebook photo upload as the hidden messages cannot be successfully extracted after the images are downloaded by the receiver. However, the mentioned image steganography techniques above can be communicated in Facebook messages and group file sharing. Google+ photo sharing, on the other hand, completely supports these five image steganographic techniques.

It was discovered from the experimental case scenarios that when social media is capable of exchanging hidden messages using image steganography, it is

a challenge for the digital forensic investigator to identify the existence of covert communication in social media, especially when the steganographic detection tool is not capable of detecting the latest steganographic signatures. Based on the process of evaluating steganography in the simulated case scenarios, an investigator guideline has been established as the output of this research project. A summary of this is shown in Figure 1.1.

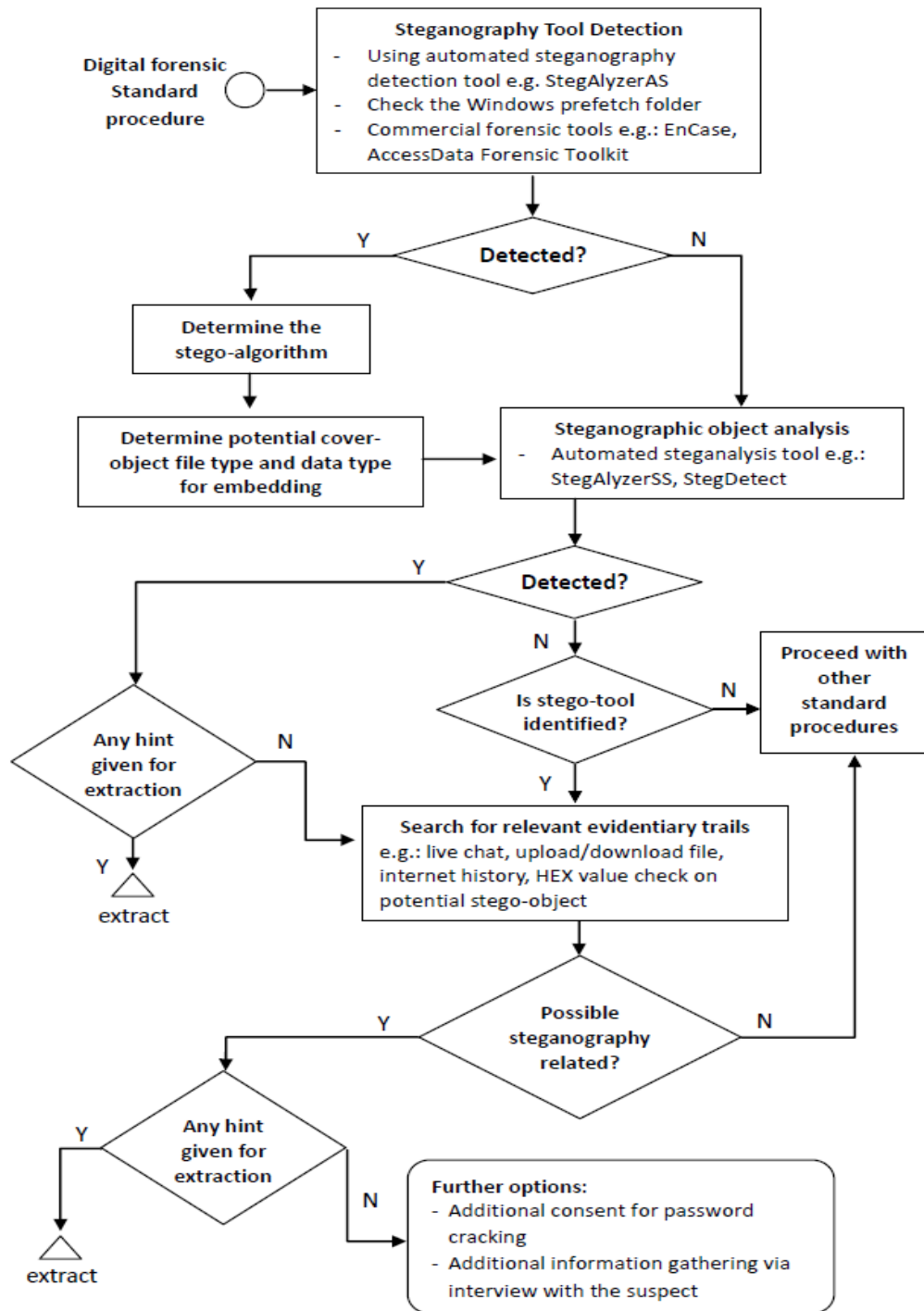


Figure 1.1 Investigator Steganographic Evaluation Flow Chart Diagram

1.4 STRUCTURE OF THE THESIS

The structure of this thesis is organized into six chapters. Chapter 1 is an introductory section where steganography is introduced and how it can be exploited in one of the most popular ways of communicating – online social networking is explained. This chapter also includes research motivation, approach, and a summary of the findings. Chapter 2 provides a comprehensive literature review of this research area to gain knowledge necessary for the thesis project and to identify the problem areas of the research.

Chapter 3 develops the research methodology for the thesis project. The objective of the chapter is to form a research method that is appropriate for the proposed project based on relevant research carried out by previous scholars and also to identify and present the data requirements and the limitations of the proposed research approach.

Chapter 4 presents the findings for each phase of the research. The variations to the research design that were encountered during the actual experiment are outlined in the beginning of Chapter 4 to highlight the changes that were necessary to obtain the findings. Any changes to case scenario, data collection, data processing, and data analysis and presentation are illustrated accordingly. The first findings are those of the pre-test result which show the steganographic techniques that are supported or restrained by the chosen online social networks. The second part of the research findings present the results of digital forensic examination and analysis from the two simulated case scenarios. The research findings are presented in table form, screen shot figures, as well as in a journal format. All results can be located in the Appendix section at the end of the thesis.

Chapter 5 is the research discussion. This chapter answers the research sub-questions, tests the research hypotheses and ultimately answers the research's main question. The hypotheses can either be accepted, rejected or indeterminate based on the supporting arguments made for and against the asserted hypothesis in Chapter 4. The research question is answered and justified according to an evaluation of the main hypothesis and the answers to the research sub-questions. The end of the chapter is a critical reflection on the project, where the findings discovered in the project are reconciled with the literature review conducted in

Chapter 2 and ends with a recommendation on steganographic evaluation that can aid forensic investigation related to in the research context.

Chapter 6 is a conclusion based on the entire research project. The problem areas identified in Chapter 2 and the research methodology are summarized. The research findings in Chapter 4 and the discussion in Chapter 5 are wrapped up and the gaps in the research findings are identified. Chapter 6 gives guidance for future research that could be considered to fill the gaps identified in the discussion of the findings.

An appendix section is included at the end of the thesis. The research appendices include additional findings as well as details of the collected data which support the research findings.

Chapter 2

Literature Review

2.0 INTRODUCTION

The objective of this chapter is to establish an in-depth understanding of steganography and to identify possible image steganographic techniques that can be utilized and applied in online social networks (OSNs). The identification of the techniques is to highlight the risks in this area so that an effective guideline for evaluating steganographic investigation can be established. There are two main areas of focus in the literature review, the first is steganography, where fundamental and related tools and techniques will be defined and the second is digital forensic investigation processes relating to the topic.

The review consists of six sections. Sections 2.1 to 2.3 discuss the development of steganography, past and present, its classification, capability, and how images can be manipulated for steganography. Section 2.4 defines OSN photo sharing capabilities, constraints, and how OSN processes assist or inhibit the uploading of steganographic photos. Section 2.5 reviews the digital forensic process, which potential sources of evidence can be gathered through social network and web browser forensics and how steganography can be identified and secret messages extracted using steganalysis. And lastly, Section 2.6 discusses the prospective problem area and the issues that are identified in the literature review which have potential for research.

2.1 STEGANOGRAPHY OVERVIEW

“Steganography is an ancient discipline which usually refers to hiding information within information” (Engle, 2003, para.3). The first recorded use of steganography was back in 440 BC when Herodotus told a slave to carry a secret message tattooed on his scalp to the Ionian city at Miletus. In order to conceal the secret message, the slave had to wait until his hair grew back before travelling to the appointed city to deliver the secret message. On arrival, he shaved his head

and revealed the secret message to the intended recipient -Aristagoras, which asked him to start a revolt against the Persian king (Fridrich, 2010).

Another classic steganography technique introduced by Giovanni Porta in the 1550's was to hide a message inside a hard-boiled egg. Porta mixed alum and vinegar to create an ink and wrote with it on the egg shell. The ink then penetrated the egg shell and left the written message on the surface of the egg's albumen. The message cannot be read until the shell is removed (Kipper, 2004). Linguistic steganography, hiding messages in text, was also a well known method used in ancient times. One of the most famous examples was by Boccaccio where he "encoded three verses (more than 1500 letters) into the initial letters of the first verse of each tercet from other poems" (Fridrich, 2010, p.4).

In ancient Chinese history, during the Yuan dynasty, the leader of a rebellion decided to secretly distribute the attack plan to his members during the Moon Cake Festival. It is Chinese tradition to serve moon cakes during the festival. The attack plans were baked into the moon cakes and distributed to the rebels on the day of the festival.

Throughout history, various methods of steganography have been created and used; the basic principle being "to communicate secret messages without making it apparent that a secret is being communicated" and this has remained unchanged to today (Fridrich, 2010, p.47). Modern steganography transforms the techniques of ancient steganography which used physical objects and hand written text by using electronic media, which hides secret messages within digital images, text, audio, video, disk space, and networks / protocols. The cover-object mainly serves the purpose of a disguise for the secret messages. This is also called digital steganography. Among all of them, image steganography is the most common and widespread applications today (Fridrich, 2010; Kessler, 2004a; Kipper, 2004).

The steganography mechanism consists of a cover-object, a secret message, an embedding algorithm, an extraction algorithm, a stego-key, and a transporting channel. The stego-key is similar to a password that is used to embed a secret message into the cover-object and it is needed to extract the secret message correctly (Kipper, 2004). During the steganographic process, the secret message will first be embedded into a cover-object with an embedding algorithm and stego-key to generate a stego-object. This stego-object can then be transported via OSN, email, website, blog, etc. to the intended receiver. The receiver then extracts

the secret message using the extraction algorithm and stego-key. A steganographic system can be summarized as in Figure 2.1 (Fridrich, 2010; Por & Delina, 2008).

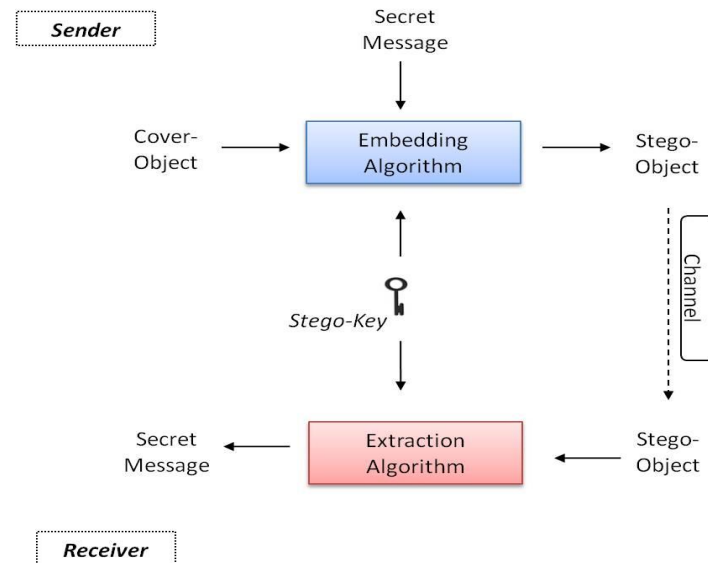


Figure 2.1: Steganographic system mechanism (Fridrich, 2010; Por & Delina, 2008)

There are three major steganographic techniques employed in digital steganography: injection, substitution, and generation of new files (Ashok, Raju, Munishankaraiah, & Srinivas, 2010; Kipper, 2004). Injection can be defined as “the insertion of a message into an existing medium” (Ashok et al., 2010, p.5989). Substitution means that “normal data is replaced or substituted with the secret data” (Kipper, 2004, p.37). And lastly, new file generation means that a new file is deliberately generated to conceal the secret message (Kipper, 2004).

2.1.1 Steganography vs. Cryptography

Steganography and cryptography in information security are intended for a common objective, which is information protection from adversaries. The difference between them is the different approaches in establishing information protection. Steganography emphasizes secret communication whereas cryptography emphasizes data protection (Cheddad, Condell, Curran, & Mc Kevitt, 2010). Steganography protects information by preventing the discovery of the very existence of a communication, using an innocent-seeming cover-object to hide information, whereas cryptography protects the information by preventing an unauthorized party from discovering the contents of a communication by using an encryption algorithm, which makes it unreadable (Raphael & Sundaram, 2011).

In steganography, the system is considered a failure once adversaries are able to detect the presence of steganography in the system. In contrast, cryptography is only considered a failure if adversaries are able to decrypt and read the message (Cheddad et al., 2010; Raphael & Sundaram, 2011). Steganography is more concerned with the embedding capacity and detectability of a cover-object, whereas cryptography is more concerned with robustness against deciphering. As for the key, a stego-key in steganography is optional. Steganography can be implemented with a stego-key to provide better security, while the key is a necessary part of cryptography (Cheddad et al., 2010).

2.1.2 Steganography vs. Watermarking

Steganography and watermarking have a common concept, which is to hide information, but technically they are different (Kessler, 2004a). Watermarking is an embedding process that hides information regarding to ownership into its cover-object (Kessler, 2004a). This means that the watermark information embedded is usually related to the cover-object, whereas the embedded information in steganography is not related to its cover-object. The cover-object in steganography is to disguise the presence of the hidden information. Therefore, steganography emphasizes its invisibility whereas watermark is flexible in its invisibility where it can be either visible or non-visible (Bandyopadhyay, Bhattacharyya, Ganguly, Mukherjee & Das, 2008).

The robustness against compression, cropping or the changing of file type for watermarking is far more important than in steganography because watermarking is used to enable detection and reveal information, whereas steganography is used to evade adversaries' detection and protect the information (Engle, 2003). Capacity wise, steganography aims to achieve maximum embedding capacity in the cover-object so that more payload (secret message) can be embedded without leaving any visible distortion, whereas watermarking only needs a small amount of embedding capacity as copyright information is minimal (Bandyopadhyay et al., 2008).

2.1.3 The Prisoners' Problem

Modern steganography is always illustrated by the “Prisoner’s Problem” model, which was provided by Simmon (Fridrich, 2010). Two fictional prisoners, Alice and Bob were to perform a prison escape and they needed to communicate and plan for the escape without arousing the cell warden’s attention – Eve, who monitored communication between Alice and Bob. If Eve finds out they have exchanged messages secretly, all communication will be stopped immediately and they will be placed in solitary confinement. Therefore, they must communicate in such a way that Eve will not suspect there being a secret message in their communication. This is the basic principle of steganography, where an outside observer is not able to distinguish whether a communication is normal or it holds hidden messages (Fridrich, 2010; Kipper, 2004). For example, Alice may draw a picture of a blue cow under a sun and give it to Eve to deliver to Bob. When Bob sees the blue cow and sun, only Bob knows the exact meaning of the object and the colour used whereas Eve may think that it is just abstract art and therefore pass it along to Bob (Kipper, 2004).

In the prisoners’ problem there are two objects of interest. Eve can be either a passive warden or an active warden in monitoring the communication between Alice and Bob. A passive warden means that Eve is only allowed to examine the picture using necessary tests to identify the existence of steganography. If Eve is an active warden, however, she may change the colour of the sun or may draw an additional object into the picture to change the original meaning of the drawing. To illustrate the implication of this in digital steganography, an active warden may apply cropping, compression, or resizing, in the transportation channel. Any of these actions could destroy the hidden message and thus disable recovery by the intended recipient. This is also a good way to prevent the use of steganography. On the other hand, Eve may choose to observe and try to learn the stego-key that is used between the prisoners and thereby try to extract the secret message so that she knows the whole story of the escape plan. Eve may even exploit the stego-key and impersonate Alice to communicate with Bob or vice versa to extract the secret message. In steganography, the act of identifying and extracting a secret message is called steganalysis.

2.1.4 Steganography Classification

Kessler (2004a), in his publication, indicated that steganography can be arranged into classifications. The two main categories of steganography are linguistic and technical steganography. Linguistic steganography is the manipulation of language or visual objects to hide secret information. It can be further divided into semagrams and open codes. Although these methods belong to linguistic steganography, in the digital world, these methods can be easily utilized without needing a complicated embedding or extraction algorithms.

In contrast, technical steganography is “a scientific way to hide secret message” (Kessler, 2004a, para. 5) such as the use of invisible ink, microdots and other size-reduction methods. Kipper (2004, p.47) elaborated that technical steganography is “the method of steganography where a tool, device, or method is used to conceal the message”. He also mentioned that technical steganography “does not necessarily deal with the written word even though it communicates information” (Kipper, 2004, p.47). This is quite true when we look at today’s steganography applications which allow any binary file to be hidden into any other binary file (Kessler, 2004b). Further illustration of different steganographic practices will be discussed in Sections 2.1.4.1 to 2.1.4.7, which cover semagrams, open codes, spam mimics, digital media, disk space, protocol, and other files. Figure 2.2 shows a steganographic taxonomy.

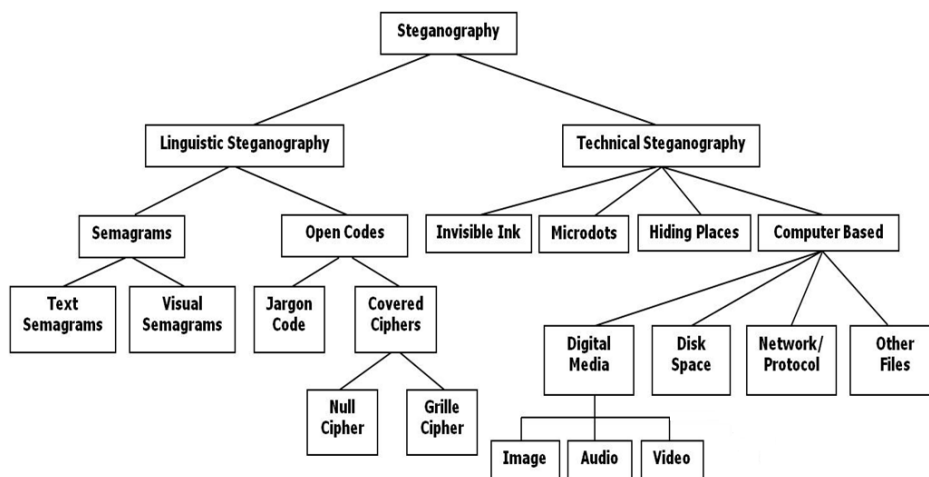


Figure 2.2: Steganographic Taxonomy (Adapted from Kessler 2004a; Kipper, 2004; Cheddad et al, 2010)

2.1.4.1 Semagrams

Semagrams hide information through the use of symbols or signs to communicate the desired message. This technique can be adapted easily digitally with the convenience of digital camera or video. For example, a sender may choose to convey a secret message by taking a photo of a pencil on the desk in an upward position indicating “attack tomorrow” or a photo showing a person’s left hand holding onto his or her right arm indicating “run as soon as possible”. In order to understand the hidden message, both sender and receiver have to share the same algorithm; a set of rules that convey the meaning of the object’s position (Kessler, 2004a; Kipper, 2004).

Semagrams also include **text semagrams**, which hide a secret message by modifying the font size or font type of the cover-text, or by adding white space into the cover-text (Fridrich, 2010; Kessler, 2004a; Kipper, 2004). One of the most well known applications that used the space insertion technique was called SNOW. It inserted the secret message into the white space at the end of each line through the use of spaces and tabs (Por & Delina, 2008). To decipher it, the receiver just need to copy and paste the text received from the sender into the SNOW application and executes the extract process, secret message can then be revealed.

2.1.4.2 Open Codes

“Open codes hide a message in a legitimate carrier message in ways that are not obvious to an unsuspecting observer” (Kessler, 2004a, para.8). This means that the secret message is actually “camouflaged” in the cover-object. Open codes are commonly used in text steganography. For example, by taking the first letter in each word from the paragraph below the secret message can be revealed as *Newt is upset because he thinks he is President*. This is called **null cipher**, a type of open code.

“News Eight Weather: Tonight increasing snow. Unexpected precipitation smothers eastern towns. Be extremely cautious and use snowtires especially heading east. The highways are knowingly slippery. Highway evacuation is suspected. Police report emergency situations in downtown ending near Tuesday” (Kipper, 2004, p.9).

Another type of open code is **Jargon code**. Jargon code is a language only a specific group of people can understand and decipher (Kessler, 2004a; Kipper, 2004). One of the commonest examples is the jargon code people use for instant messenger chat or comments left on blogs or social networking websites: LOL = laugh out loud, BRB = be right back, FYI = for your information, GTG = got to go, and so on.

Grille cipher was invented by Girolamo Cardano. To encode the secret message, the sender first randomly punched slots in a piece of cardboard that aligned with writing lines to create a “grille”. The grill was then put over a piece of paper and the secret message was written in the slots. After that, the grille was removed and the fragments of text were filled in to create an innocuous cover-text that may look like a regular letter or note. To read the secret message the recipient has to use a piece of cardboard that has the same punched slots (Kessler, 2004a; Kipper, 2004).

2.1.4.3 Spam Mimic

Spam mimic is a good example of a new file generation technique that is used in digital steganography. It is also considered a type of **null cipher** (Kessler, 2004a). It transforms the secret message into a spam-like message (Figure 2.3) normally found in email inboxes. The spam message does not really make any sense and would be disregarded by others except the intended receiver, as it looks like a nuisance spam email. It also manages to deceive the email filter and successfully transporting the message to the intended receiver. Spam mimic can be found at www.spammimic.com. The sender just has to key in a short secret message and the website will encode the secret message into a text block that looks like spam. This grammar-based mimicry function was proposed by Peter Wayner (Kessler, 2004a). Figure 2.3 is an example of a spam message created using spam mimic with the secret message “*attack@1400*”. The sender copies and pastes the generated spam message (stego-text) into their email or OSN’s message and send it to the intended receiver. To reveal the secret message, the receiver again copies and pastes the spam message into the spam mimic website, uses the decode button and they are able to recover the secret message (Fridrich, 2010; Kessler, 2004a; Kipper, 2004; Newman, 2007).

Dear Professional , Your email address has been submitted to us indicating your interest in our newsletter ! This is a one time mailing there is no need to request removal if you won't want any more ! This mail is being sent in compliance with Senate bill 2516 , Title 9 , Section 303 . Do NOT confuse us with Internet scam artists . Why work for somebody else when you can become rich in 16 days ! Have you ever noticed most everyone has a cellphone and more people than ever are surfing the web ! Well, now is your chance to capitalize on this ! WE will help YOU turn your business into an E-BUSINESS and increase customer response by 120% ! You are guaranteed to succeed because we take all the risk . But don't believe us ! Ms Simpson who resides in Hawaii tried us and says "I've been poor and I've been rich - rich is better" . We assure you that we operate within all applicable laws ! We implore you - act now ! Sign up a friend and you get half off . Thanks .

Figure 2.3: An example of a spam mimic message generated using the web-based tool provided by www.spammimic.com

2.1.4.4 Digital Media

Information embedded into the cover-object using least significant bit (LSB) substitution is the most common steganography technique applied to digital images, audio and video. This technique was first used in digital images by Kurak in early 1990. He showed how to utilize the LSB of an image to hide another image (Potdar, Khan, Chang, Ulieru, & Worthington, 2005). LSB works by substituting the low-order bits of image data, which is the 8th bit in a byte of a cover-object, with a bit of the secret message. The advantage of utilizing digital multimedia files as cover-objects is due to their enormous amount of redundancy. Redundancy was defined by Morkel, Eloff, & Olivier (2005, para.12) as “the bits of an accuracy far greater than necessary for the object’s use and display”. Moreover, the redundant bits, also considered “noisy” areas, deal with natural colour variation and are hardly detected by the human eye (Johnson & Jajodia, 1998). A similar approach was further researched and employed on audio and video files, which are capable of accommodating a higher payload without any affect on playability. A further discussion of image steganographic techniques will be presented in Section 2.3.

2.1.4.5 Disk Space

Allocated unused disk space, which is also called slack space, can be used to hide information. Allocated unused space is created when the operating system is saving a file and it allocates a minimum cluster to store that file, for example 32KB, even though the actual data is only 12KB, and the file requires less than its allocated space, the entire cluster is reserved for that particular file. Therefore, the extra unused space, which in this case is 20KB, can be used to hide information without it showing up in any directory or file system as from the point of view of the operating system, the entire cluster is already occupied (Kipper, 2004). Additional information on data hiding in disk space can be found in Berghel, Hoelzer, & Sthultz (2006).

2.1.4.6 Protocol

Protocol steganography is a method of “embedding information within messages and network control protocol used in network transmission” (Bandyopadhyay et al., 2008, p.110). The TCP/IP in the network layer can be used as a covert channel to transmit a data packet between hosts. For example, information in a TCP/IP header can be manipulated into ASCII values for transmission to an outside source. Other types of network information or protocol that can be used to hide information can be ICMP packets, routing control information, or user datagram protocol (UDP) datagrams (Newman, 2007).

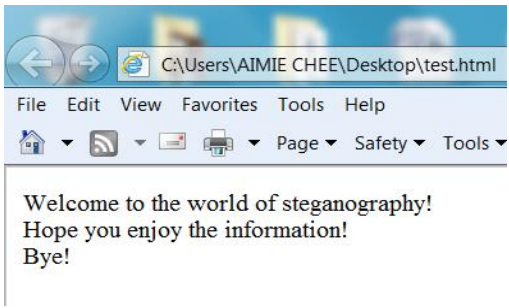
2.1.4.7 Other Files

Simple information hiding can also be performed in a Word document, PowerPoint, or other file formats. For example, an image or text block can be hidden under another image in a PowerPoint file. A text block can also be hidden by matching its font colour to the background colour to disguise its existence. Another way to hide a message is in Microsoft Word, Excel, PowerPoint file properties details. Messages can also be hidden in the macro function that is available in Microsoft Office.

The HTML (Hyper Text Markup Language) coding used to publish web pages on the internet can also be used to embed secret information, but this is an unsafe method as the message is visible in the code itself although it will not be seen until the receiver looks at the source code. An example of this is using the

comment code in html to embed the information. The comment code is normally used by the developer to leave a note in regard to the code, and this comment is ignored by the web browser and will not be published on the web page (Newman, 2007). As depicted in Table 2.1, the comment line ‘This can be used to hide information’ is not displayed in the actual webpage.

Table 2.1: Information hiding using HTML

<u>Source Code:</u>	<u>Webpage Display:</u>
<pre> <html> <body> <head> <title>Welcome!</title> <!--This can be used to hide information --> Welcome to the world of steganography!
 Hope you enjoy the information!
 Bye! </body> </html> </pre>	

2.2 DIGITAL IMAGE FORMAT

Image steganography, as stated earlier in Section 2.1, is the most popular method among of digital steganography. It exploits the weakness of the human visual system (HVS) by modifying colours in an image which cannot be easily detected. The image that we see on the computer screen is usually in a grid form displayed horizontally, row by row (Morkel et al., 2005). This is visible when we magnify the image on the screen. There are a few common digital image formats that are widely used such as raster, palette, transform, and vector. However, the transform format - JPEG is the most common and popular digital photographic format that we see today (Fridrich, 2010). JPEG not only provides small data size; it is also capable of achieving “close approximations to high quality digital photographs” (Johnson & Jajodia, 1998, p.27). To understand image steganograpy it is necessary to review how colour is digitally represented and the common formats are used to store digital images in the following sub sections before a further discussion of how it can be used for image steganography is given.

2.2.1 Colour Representation

The human visual system perceives colours by light intensity and is limited to a small subset of all possible colours (Fridrich, 2010). An image on a computer screen is actually a grid formed from the numeric representation of colours with each dot of colour referred to as a pixel (Morkel et al., 2005). The variation of colour in an image that the human visual system perceives on the computer screen is actually derived from the light intensity of the basic primary colours of red, green, and blue (RGB). All other colours derived from RGB are called secondary colours. Morkel et al. (2005, para.18) stated that “to a computer, an image is a collection of numbers that constitute different light intensities in different areas of the image”. This light intensity is the pixel and is represented by bits. In RGB, each component is from a range between 0 – 255 and each component intensity can be represented by an 8-bit integer. Figure 2.4 shows that when red and green are at 0 (the lowest intensity) and blue is at 255 (full intensity), a primary blue colour is presented. When all the three colours are at full intensity, white is formed. When they are all at the lowest intensity, black is formed (Fridrich, 2010). Basically, by varying the red, green, and blue intensities any other secondary colour can be generated.

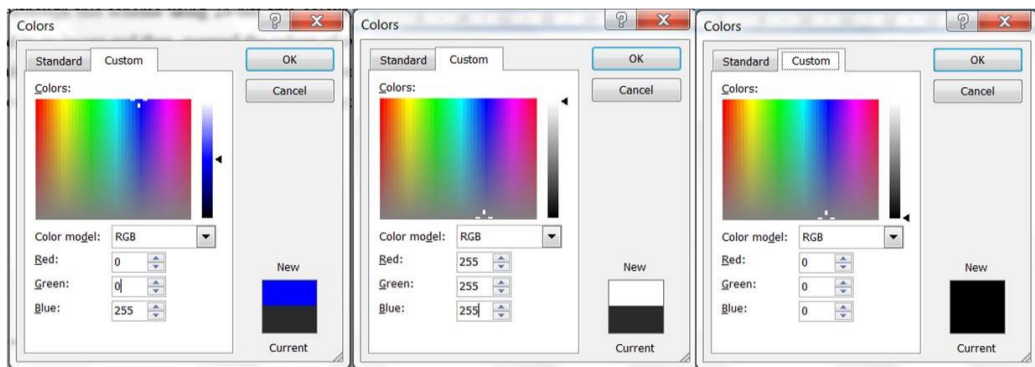


Figure 2.4: RGB colour intensity representations

Although the RGB colour model is readily perceived by the human visual system, it is considered redundant as the RGB signals are highly correlated to themselves, which makes the transmission uneconomical. Therefore, a new colour system was developed, called the YUV model (Fridrich, 2010). The Y component is the brightness, also called luminance, while the U and V components are for colours, called chrominance (Hamid, Yahya, Ahmad, & Al-Qershi, 2012). The intensity

range of YUV is different to RGB, especially the U and V components where the range is in between -179 and 179, which enables YUV components to be represented by 8 bit integers. YUV was further modified into the $YCbCr$ colour model so that it could be used for digital as a format (Fridrich, 2010).

2.2.2 Raster Format

The number of bits used in each pixel in an image can vary depending on the image format and the number of bits allocated per pixel (Fridrich, 2010). In the raster format, the digital true colour image is normally stored in a 24-bit file that derives from the RGB colour scheme. Each primary colour is represented by 8 bits, which means that there are 3 bytes or 24-bits to represent a colour in a pixel and in each pixel there can be 256 quantities of red, green, and blue that can add up to more than 16 million combinations, and therefore can create more than 16 million colours (Fridrich, 2010; Hamid et al., 2012; Johnson & Jajodia, 1998; Morkel et al., 2005). In addition, the raster format usually uses lossless compression to decrease the amount of image data that needs be stored.

8-bit image files however use 8 bits (1 byte) to represent colours in a pixel. Obviously, the colour combinations for 8-bit files are limited, and only 256 different colours are able to be displayed. Obviously, 8-bit image files' size will be smaller than the 24-bit files. For example, an 8-bit image with 320 x 240 pixels will have 76800 bytes (76.8KB), whereas a 24-bit image file of the same dimension will have 230400 bytes (230KB). 8-bit files are usually found in gray scale images where the 8 bits are utilized to represent 256 different shades of gray. As for monochrome pictures, they need only 1 bit per pixel with only black or white to be displayed. Image files such as BMP (Bitmap), TIFF (Tagged Image File Format), and PNG (Portable Network Graphics) are file types that render using the raster format (Fridrich, 2010; Hamid et al., 2012). Of all of these, the BMP creates the largest file sizes and thus has a larger capacity for secret message embedding, but it is ill-suited for use on the internet as it needs a higher network transmission capability (Cheddad et al., 2010).

2.2.3 Palette Format

The other way to utilize 8-bit image files is by using a palette format. In a palette format, the image consists of three attributes, the header, the image palette, and the image data. The palette is able to store 24-bit colour but is limited to 256 colours only (Fridrich, 2010; Kessler, 2004a). If an image has more than 256 colours, then a palette will be created then each pixel of the colour will be converted to a palette colour. This, therefore, limits the unique colour representation of an image and it usually shows signs of degradation in an image. However, a degraded image is advantageous for secret data embedding, as the noise in the image is a good cover up as it draws less attention (Johnson & Jajodia, 1998). The palette format is usually used for images that do not required great colour depth, such as cartoons, logos, or line drawings. Lossless compression is also employed in the palette format. The most common image file type that uses the palette format is GIF (Graphics Interchange Format) (Fridrich, 2010).

2.2.4 Transform format – JPEG

While raster and palette image formats (also called spatial-domain formats) use pixel by pixel encoding, the JPEG (Joint Photographic Experts Group) image format is classified as a transform-domain format. Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) are the most commonly used transforms for generating a .jpg/.jpeg format file (Fridrich, 2010). The transform is needed to achieve JPEG compression in order to provide a high colour quality image with a smaller file size as compared to the spatial-domain formats (Johnson & Jajodia, 1998). JPEG compression is a lossy compression because the decompressed image (image that is viewed) will not be identical to the original image after compression (Fridrich, 2010).

According to Fridrich (2010), there are five steps needed in order to save an image in the JPEG format. First, the RGB colour model is transformed into the $YCrCb$ model. Second, since it has been proven that human eyes are insensitive to minor changes in colour, but are very responsive to brightness changes, JPEG compression utilizes this weakness of the human visual system by down-sampling the colour component, C_r and C_b , to achieve a higher compression ratio and divides the luminance, into 8 X 8 pixel blocks. Third, each block of $YCrCb$ signals

are transformed into a frequency domain with the DCT from the spatial domain. This process is done by sorting the pixels into 8 X 8 pixel blocks and transforming the pixel blocks into 64 DCT coefficients that approximate the luminance and chrominance of a block (Fridrich, 2010; Hamid et al., 2012; Kessler, 2004a; Morkel et al., 2005). According to Morkel et al. (2005, para.38), the DCT transformation process is similar to converting “the pixels in such a way as to give the effect of ‘spreading’ the location of the pixel values over part of the image”. Fourth, the coefficients in the blocks after the DCT transform are then quantized. This is called the quantization step. In this process the DCT coefficients in a block are divided by an integer value and rounded up to the nearest integer (Fridrich, 2010). This step again exploits the human visual system’s weakness. Human eyes are capable of differentiating brightness changes over a relatively large area, but incapable of distinguishing the “distinction between different strengths in high-frequency brightness” (Hamid et al., 2012, p.173). Thus, quantization is to further reduce the strength of higher frequencies without making any apparent changes to the image. Lastly, the quantized DCT coefficients are encoded using bits and then losslessly compressed with Huffman or arithmetic coding to generate an output file with a ‘.jpg’ or ‘.jpeg’ extension (Fridrich, 2010).

In order to view the JPEG image file (also called decompression), the above mentioned steps have to be reversed so that the spatial domain representation of the JPEG file can be obtained (Fridrich, 2010; Kipper, 2004). Kipper (2004, p.50) stated “during the decompression, JPEG recovers the quantized DCT coefficients from the compressed data stream, take the inverse, and displays the image”. Quantization, however, is irreversible (Fridrich, 2010).

2.3 IMAGE STEGANOGRAPHY

The extensive use of digital images and the high amount of redundant bits in digital images have encouraged the use of digital images as cover-objects for hiding secret messages (Morkel et al., 2005). According to Cheddad et al. (2010), the most popular image formats found on the Internet today are GIF, JPEG, and PNG. In this section, various steganographic techniques will be reviewed and the free tools that can be downloaded freely and readily from the Internet will be

focussed on. Furthermore, the review of steganographic techniques will be based on image formats that are acceptable on OSNs.

2.3.1 Text File (.txt) Injection into Image File

Cheddad et al., (2010) demonstrated a simple steganographic technique that does not require a high level algorithm which appends text into the end of file (EOF) of an image file using a Windows DOS command line. It can be easily performed by typing ‘copy /b cover_object.jpg + secret.txt stego_object.jpg’ into the command prompt according to the directory of the file located. Basically the command is copying the text data from a text file, inserting it after the EOF tag of the cover_object.jpg file (Figure 2.5) and generating a new image file that has the inserted text (Figure 2.6). The advantage of this method is that it does not affect image quality and therefore it cannot be visually identified when comparing the two images. Furthermore, the image histograms for both cover-object and stego-object are identical as this method hides the secret message after the EOF tag. To reveal the secret message, the intended recipient can open the stego-object by using a notepad application and the secret message can be found at the bottom part of the page (Figure 2.7).

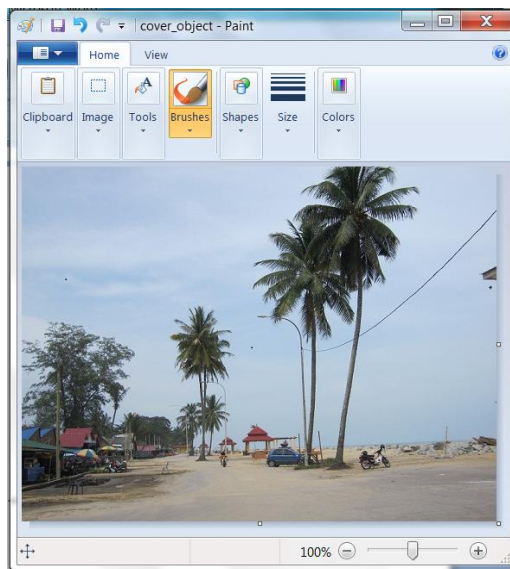


Figure 2.5: Cover-object

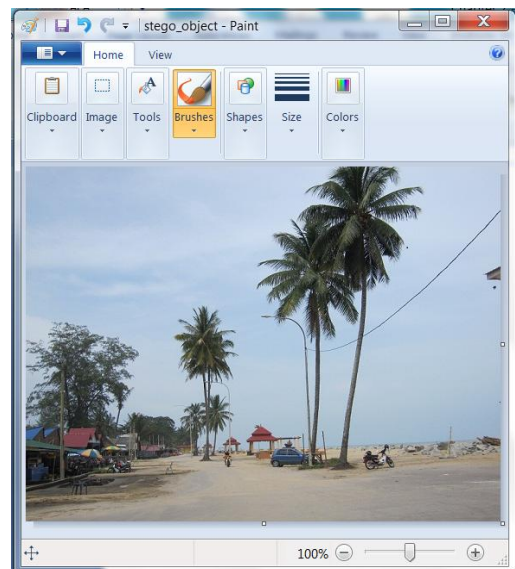


Figure 2.6 Stego-object

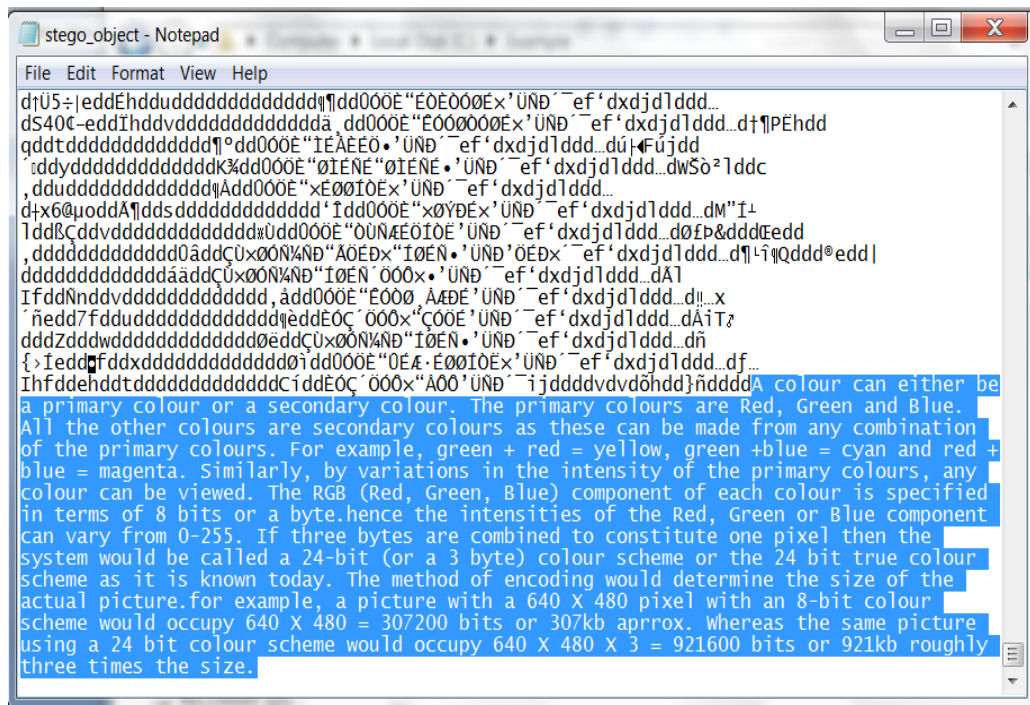


Figure 2.7: Secret message revealed through notepad application.

Although this method exploits the image file for steganography in a simple way, it does have its drawbacks. One significant issue with the stego-object is its file size which it will be larger after the injection. Usually, the file size of the generated stego-object is the sum of both file sizes. Another issue is that this method is not resistant against any kind of active attack including editing, resizing, cropping, and so on (Cheddad et al., 2010).

2.3.2 Zip File (.rar / .zip) Injection into Image File

There is another similar DOS command that can be used to perform steganography in an image file, which is 'copy /B cover_object.jpg + secret.rar stego_object.jpg'. The difference between this command line and the one in Section 2.3.1 is the secret messages file type. In this command line a zip file (.rar) is incorporated instead of .txt file. The benefit of using the zip application is that it enables the inclusion of any type of binary file into the cover-object. For example, the sender has three secret photos that he or she wants to send to the intended receiver, so the sender first zips the three photos into a .rar file using the WinRAR application (for Windows 7). Then, the sender can utilize another innocent image file (cover-object) to enclose the zip file with the provided command line to compact the two files into one innocent-looking stego image. When the intended

receiver receives the stego image (stego-object), he or she is able to retrieve the secret messages by using the WinRAR application (Windows). Those who do not know the protocol may only see the stego-object as a regular .jpg image file (Trapani, 2007). Similar to Section 2.3.1, this technique does not resist any kind of active attack, but it has an advantage over the technique that shown in Section 2.3.1. If similar secret messages were appended to an image file, one using a text file and the other using a zip file, the text file (secret data) appended to an image file can easily be seen in a HEX editor (Figure 2.8) and can be read by a text editor, whereas if it were incorporated with a zip file, the text file (secret data) is compressed and therefore unable to be read in either the HEX editor (Figure 2.9) or in the text editor until it is recovered using the right zip application (Figure 2.10).

Trapani (2007) said that the reason why this method can be used is because image data was stored in the header while zip file data is stored in the footer. Thus, when the image is viewed it only displays those bits before the EOF; anything after the EOF will be ignored (Cheddad et al., 2010). Whereas in ZIP files, BOF (beginning of file) or EOF tags do not exist thus, when a zip application opens a stego-object that has zip files within it, it only searches for a zip central directory that is recognized by the application and recovers the files that are stored in the zip file ("Zip (file format)," 2012).

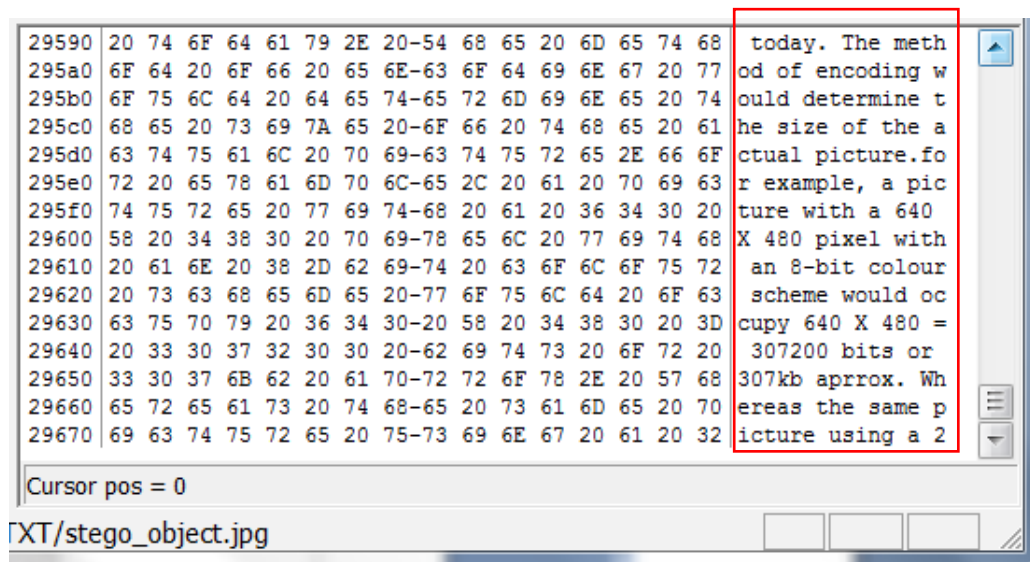


Figure 2.8: Stego-object produced by appending a text file (.txt) to an image file

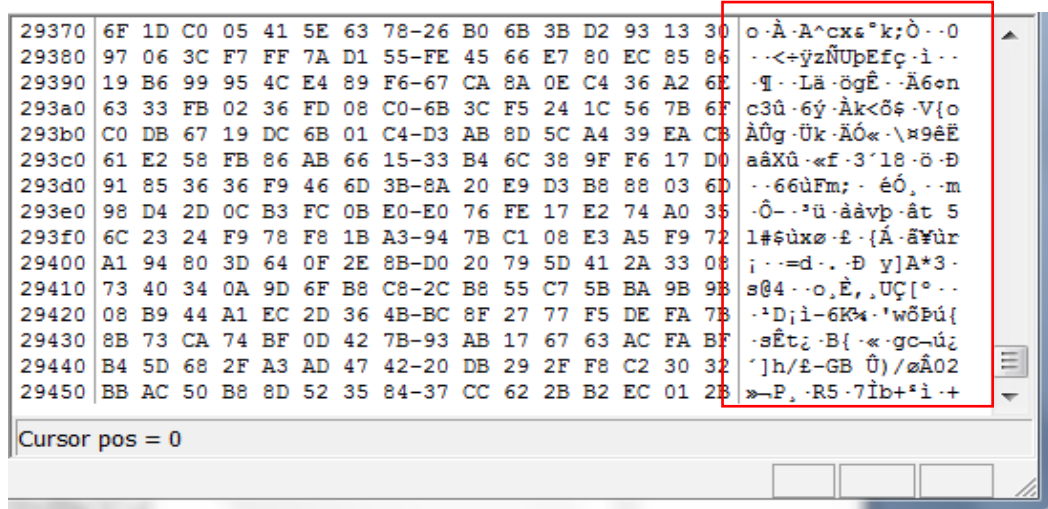


Figure 2.9: Stego-object produced by appending a zip file (.rar) to an image file

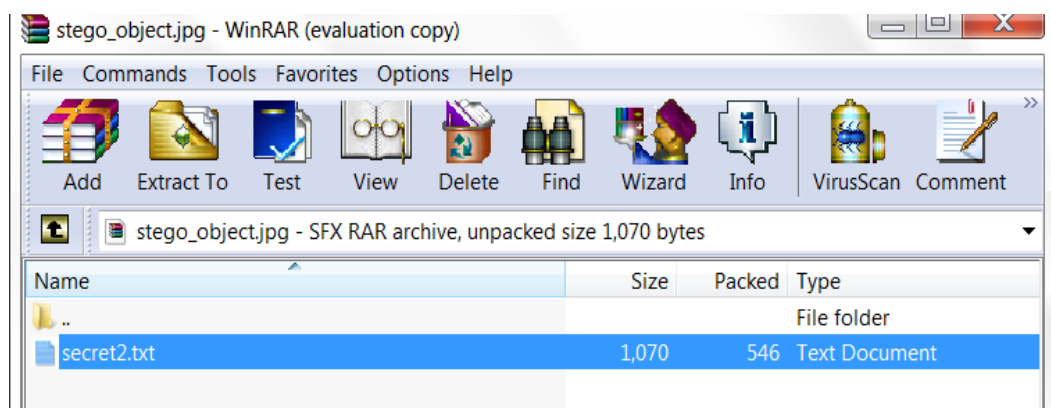


Figure 2.10: Secret data extracted using the WinRAR application

2.3.3 Hiding in EXIF

Another simple steganography that can be used is by way of hiding secret messages in the EXIF (Extended File Information) file (Figure 2.11). EXIF is usually used to store information in regard to image data, camera manufacturer, or other file details. It is the image's metadata information that is located in the header of the file. This technique can be done easily by right clicking on the image file and choosing the properties option. In the details tab, the sender can type their secret message in any of the value columns that allow text inputting. To read the secret message the receiver has to know the secret message hiding protocol and, by following the same steps, he or she will be able to find the secret message. This header can be easily exploited to include other messages as EXIF data is usually ignored (Cheddad et al., 2010). Although this is not a secure and reliable method and has the same weaknesses as the previous methods in 2.3.1 and 2.3.2,

it is a practical method that should not be ignored when dealing with steganography.

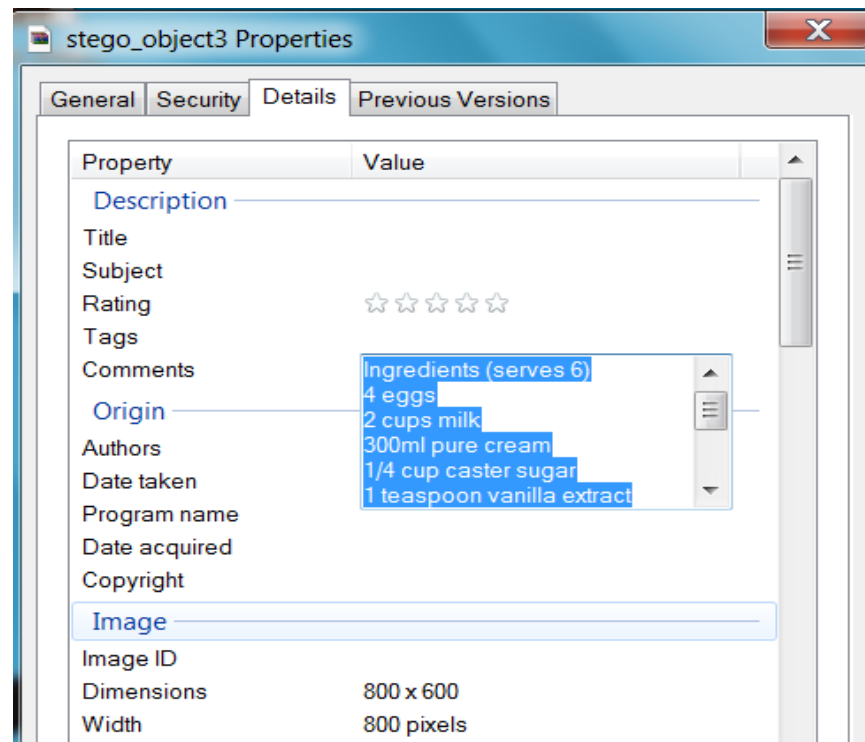


Figure 2.11: A secret recipe was hidden in an image's EXIF properties.

2.3.4 Least Significant Bits (LSB) Substitution in Spatial Domain Images

Steganography by LSB substitution “replaces redundant or unneeded bits of a cover with the bits from the secret message” (Kipper, 2004, p.39). For example, a combination lock password which is ‘213’ needs to be hidden in an image. When implementing the LSB technique, it replaces the right most bit of a colour with a bit from the secret message. In this case, the binary number for 213 is 11010101. In order to embed each bit of 213 in an image, 8 bytes from the image (cover-object) is needed, as only 1 bit of least significance will be used to embed 1 binary number of the secret message, so that it will not visibly distort the cover-object. The 8 bits that make up a byte go from left to right in the order of importance to represent a colour value, for example 01001100. Changing the most significant bit (MSB) – 0, which is the left most bit to ‘1’ will drastically change the colour. However changing the right most bit – 0, also called the least significant bit (LSB), to ‘1’ will have little effect on the colour it represents. Furthermore, this change is hard for human eyes to detect as each RGB colour component has 256 possible

intensities and LSB substitution only slightly modifies colour intensity (Morkel et al., 2005). For a cover-object that is a 24-bit image, each LSB bits of RGB colour component will constitute 3 bits of secret message that can be embedded in a pixel. Therefore if an image's size was 480 X 320 pixels it could embed up to 460800 bits of secret data presuming that every single byte in an image data was used to hide the secret message. Additionally, if an image with high colour variation had been carefully selected; the secret message could even be hidden in the second least significant bit or more without visual distortion (Johnson & Jajodia, 1998; Morkel et al., 2005). Table 2.2 shows an example of how the combination password of '213' is embedded into the partial bytes of a cover-object. As can be seen in the table, of the 8 bytes of the original image, only 5 bytes are changed to represent '213' (Bandyopadhyay et al., 2008; Kipper, 2004). This situation is normal, as according to Morkel et al. (2005), only half of the bits in the entire image are used to hide the secret message on average.

Table 2.2: LSB Substitution Table

Partial bytes of an original Image	Secret message Bit – '213'	LSB Substitution on the original Image
10000100	1	1000010 1
10000110	1	1000011 1
10001001	0	1000100 0
10001101	1	1000110 1
01111001	0	0111100 0
01100101	1	0110010 1
01001010	0	0100101 0
00100110	1	0010011 1

Although the large size of BMP images is very favourable for steganographic purpose, it is not such a desirable and common format to use on the Internet as its large size takes longer to load into a web browser (INFOAVE, 2011; Morkel et al., 2005). However, this is no longer such a big issue with current high speed Internet technology. It is not even a consideration if the images are to be transmitted through an OSN as basically the OSN is a personally managed web content site. As long as the image file is of reasonable size and in an acceptable file format to the OSN, it can be transmitted without worrying about speed. The only reason it would raise a red flag nowadays is its infrequent use on

the Internet as compared to JPEG, PNG, or GIF. Nevertheless, a BMP cover-object is still a basic carrier illustrating the concept of LSB substitution in pixel spatial domain. Steganography tools that are freely downloadable on the Internet using this concept are Stegotif, Blindside, S-Tools, Hide in Picture (HIP), and so on (Kipper, 2004; Malik, 2009). Figure 2.13 shows the stego-object with a 1.04KB text file embedded using HIP. As can be seen, there are no perceivable differences with the image file in Figure 2.12 in terms of the image or the image file size.

LSB substitution was further developed and implemented into the GIF format as well, where the colour palette indices were used to embed the secret message. Steganography tools that have been developed to use GIF images as cover-objects include EzStego, Hide and Seek, GIF-Shuffle, Gif-It-Up and more (Malik, 2009). More steganographic tools available can also be found in the research conducted by Hayati, Potdar, & Chang (2007).

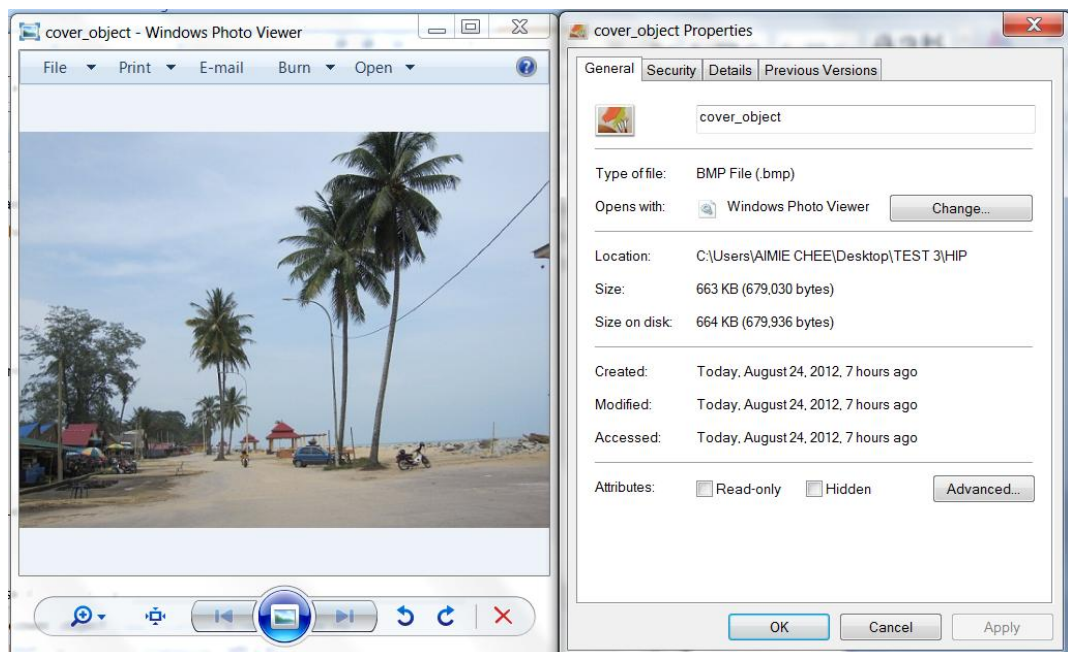


Figure 2.12: BMP format cover-object with an original size of 663KB

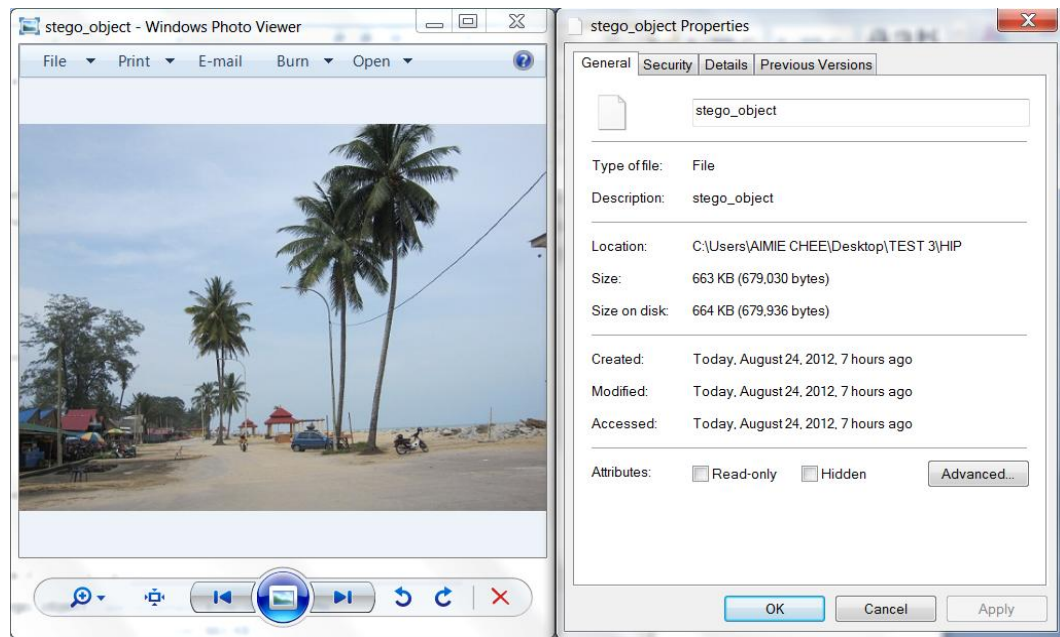


Figure 2.13: BMP stego-object that embedded with 1.04kb of secret message created using the Hide in Picture steganography tool.

2.3.5 Least Significant Bits (LSB) Substitution in DCT

A more advanced steganographic technique has evolved with the emergence of the JPEG format. In the JPEG format, DCT is used to accomplish JPEG compression. During DCT transform, the coefficient value can be modified for secret message hiding (Potdar et al., 2005). The JPEG format was at first considered to be useless for hiding information due to its lossy compression algorithm, which may destroy hidden messages, but its properties have been successfully exploited by Derek Upham, who developed the first embedding algorithm for JPEG images (Morsy, Nossair, Hamdy, & Amer, 2011). “Its embedding technique sequentially replaces the least-significant bit of DCT coefficients with the message’s data” (Morsy et al., 2011, p.172). This can be accomplished because JPEG compression algorithms are divided into lossy and lossless. The DCT and quantization steps use lossy compression whereas the final encoding part for further compression, using Huffman encoding, is actually lossless, therefore LSB substitution for secret message embedding can be done after the DCT and quantization by modifying the least significant bits of

coefficient values before the final encoding without affecting the secret message (Morkel et al., 2005). This technique is unsusceptible to visual attack as the modification is performed in the frequency domain rather than the spatial domain (Provos & Honeyman, 2001).

The steganography tools commonly used for JPEG images are JSteg, OutGuess, StegHide, JP Hide and Seek, Invisible Secret, F5, SteganPEG and so on. More information regarding the steganography tools for JPEG images can be found in Hayati et al. (2007) or Kipper (2004). Figure 2.15 is a stego-object with 1.04KB text embedded using JP Hide and Seek. Notice that the cover-object in Figure 2.14 was further compressed by JP Hide and Seek from 120KB to a smaller size of 74.2KB after embedding the 1.04KB secret message into the stego-object (Figure 2.15).

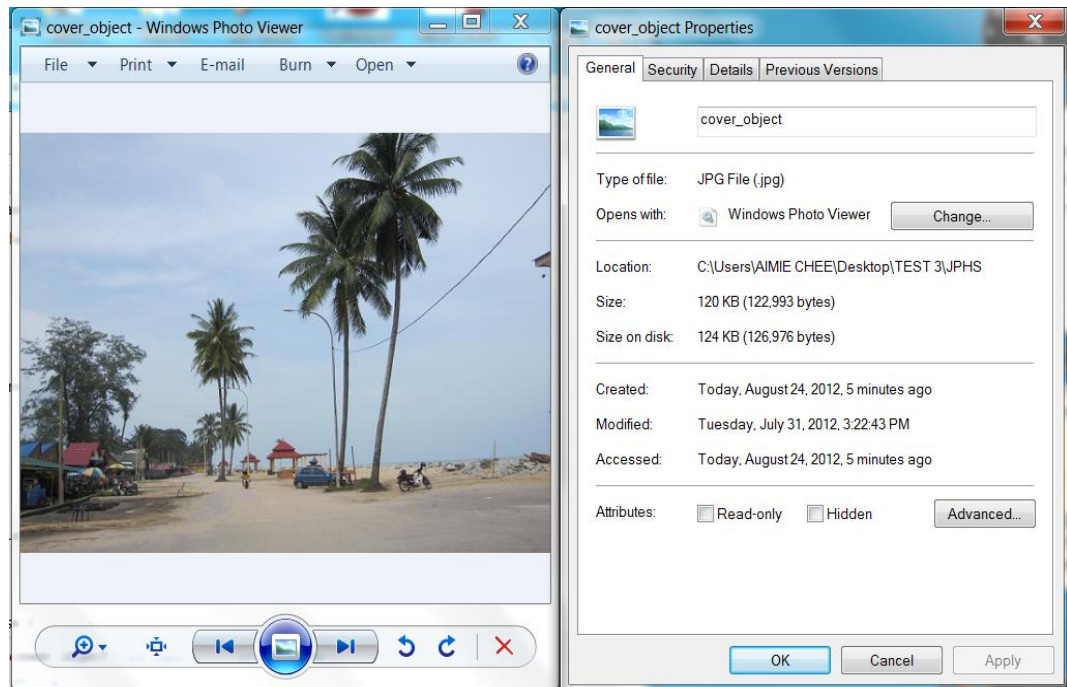


Figure 2.14: JPEG format cover-object with an original size of 120KB

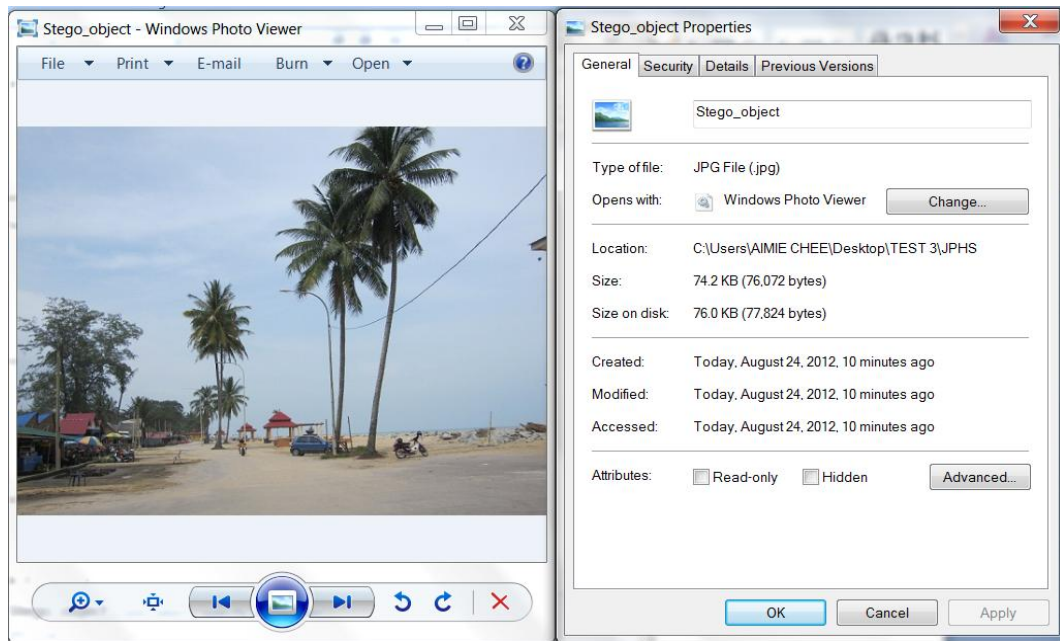


Figure 2.15: JPEG stego-object embedded with 1.04kb of secret message created using the JP Hide and Seek steganography tool.

2.4 SOCIAL NETWORK PHOTO SHARING CAPABILITIES

Free steganographic tools available on the market are capable of performing information hiding in BMP, GIF, JPEG or even PNG. As the proposed research is to examine the applicable image steganography in an OSN, it will be vital to understand how OSNs process images and what restrictions they have for photo sharing as OSNs usually have policies that constrain the size and format of an image; if the uploaded images do not meet the defined policy, images are either rejected or auto compressed, cropped, resized, or reformatted by the OSN. This modification is serious for images embedded with a secret message, as any of the modifications may destroy the hidden message as steganographic tools available on the market so far may not be robust enough to resist these active attacks.

The research conducted by Castiglione, Cattaneo, & De Santis (2011) showed that OSNs pre-processed the uploaded images before publishing them on the user's content and changed the images' characteristics. The experiment conducted was based on three OSNs: Facebook, Badoo, and Google+. Their experimental results showed that the three OSNs changed the pixel resolution and metadata of uploaded pictures to fixed values. Facebook and Badoo use pre-

defined JPEG quantization tables to compress the images. Facebook and Badoo only accept JPEG image files; any other image format will be automatically converted to a JPEG format while Google+ is more flexible; JPEG, BMP, PNG and GIF image formats are accepted for uploading without format conversion. Usually, if uploaded images satisfy the OSN's defined size and format, they will be published without resizing or reformatting. If the images are not within the defined constraint, they will be adjusted to a size and format that complies with the OSN's policies. Since steganographic messages will be destroyed by compression, resizing, and format changes, it is necessary to take this information into consideration when performing covert communication on an OSN. The image pre-processing by different OSNs when uploading is summarized in Table 2.3:

Table 2.3: OSN pre-processing activities on uploaded images

	Facebook	Badoo	Google+
Compressed image	Yes	Yes	No
Resize	Yes	Yes	only when it's over the size constraint
Format converted	Not on JPEG others will be converted to JPEG	Not on JPEG others will be converted to JPEG	No
Format accepted	JPEG	JPEG	JPEG, BMP, PNG, GIF

Referring to Table 2.3, if a steganographic image was to be posted on Facebook or Badoo, the only possible carrier for a secret message would be the JPEG format. However, the newly released Facebook service, called 'file sharing' has given option to users within a group to share a file of up to 25MB. The terms of service only mentioned that music files and .exe files are not permitted, which means any image file type can be shared via the file sharing feature and with this feature images do not have to go through regular Facebook photo upload pre-processing (Freeman, 2012). This has increased the choice of cover-object as steganography tools capable of embedding secret message into different types of image such as BMP, JPEG, GIF, PNG, and even TIFF can be used. Even though music files and .exe files are not permitted, those files can still be transmitted through steganography without notice. In Google+, JPEG, BMP, GIF, PNG are formats that can be used for image steganography. The following Subsections, 2.4.1 and

2.4.2, will illustrate how and where photos can be shared in Facebook and Google+.

2.4.1 Facebook Photo Sharing


There are a few ways that a person can share their photos on Facebook. The most common one is uploading the photos via upload Photo/Video or the create Album feature in Facebook. Both features can be found in either a person's home wall or a group wall. Once the photos have been selected (in this case, selecting the photos that have been embedded with secret information) by clicking the post button, the photos will be uploaded to the user's or group's wall. However, it has been discovered that secret messages were unable to be extracted from the downloaded steganographic images especially those steganographic images that were created by JP Hide and Seek, StegHide, F5, and SteganPEG (Castiglione, D'Alessio, & De Santis, 2011). This is due to Facebook's pre-processing compression algorithm that is applied to all uploaded photos regardless of image file size, which had destroyed the secret message. Yet, one tool has been discovered that has the ability to extract the embedded secret message in images that have gone through the Facebook compression algorithm; SilentEye developed by Chorein (2010). Although SilentEye has the capability to survive the Facebook compression, the generated steganographic image has significant distortion which is perceivable to the human eye.

The other way to share photos in Facebook is through the upload file feature in the group's wall. In order to share files within the group, the user has to first create a group with members with whom the user wishes to communicate. The upload file feature is similar to virtual storage where User A is able to upload files onto the group's wall and User B can download it later from the group's wall. For example Alice created a group named 'Dream' in Facebook and added Bob as a member of this group. Now Alice and Bob are able to communicate in the 'Dream' group. If Alice has a steganographic image to share with Bob, she can use the upload file feature in the 'Dream' group and upload the image file. To extract the secret message, Bob can download the image file from the 'Dream' group's wall and extract the secret message using the appropriate steganographic tool both Alice and Bob have agreed upon. This way of file sharing successfully exfiltrates the steganographic image and successfully transmits the secret message

without having to worry about Facebook's photo compression. With the upload file feature, steganographic images generated by any available image steganography tool can be successfully transmitted in a Facebook social network group either in an open group, closed group or secret group, which is dependent upon how Alice set the group's privacy. If it is an open group, anyone can see the group, who is in the group, and all the posts or activities of the group. When it is a closed group, anyone can see the group and the members of the group but only members can see the posts or activities. A secret group is only open to its members and only members can see the group, who is in the group, and the content of the group's page.

Sending messages is also a common activity on the Facebook social network and a steganographic image can be sent as an attachment to a message to friends in the network or to the intended recipients using the recipients' email addresses. Likewise, Facebook users can receive messages with steganographic image attachments from friends in their network or receive messages sent to their Facebook email account (e.g. user@Facebook.com) from someone using a traditional email system such as Hotmail, Yahoo Mail or Gmail ("Messages basics - Facebook help center," n.d.). For example, Alice sent a message with a steganographic image attachment to Bob, who is a 'friend' in Alice's Facebook. Alice can also send a steganographic image as an attachment to Bob's email address even though Alice and Bob are not 'friends' in Facebook. Furthermore, Bob does not need to have a Facebook account to receive a Facebook message from Alice. Similarly, Bob is able to send steganographic image attachments to Alice's Facebook's email address without having to be Alice's Facebook friend or having a Facebook account. Obviously, file attachment in the Facebook message feature is capable of facilitating steganographic distribution.

2.4.2 Google+ Photo Sharing

The photo sharing feature in Google+ is not as complex as Facebook. Google+ has a basic photo sharing feature which is the 'add photo ' function which can be found on the user home page, profile page, or the '+ Share' icon at the top right hand corner of the screen. Users can either instantly upload the photos into a selected circle's page or into a selected album. Unlike Facebook, Google+ does not pre-process the uploaded images with photo compression. If the uploaded

images are within the constraints of the uploading policy, the image will be published as it is. Google+ users can either share their photo publicly, which allows everyone who has Google+ to see and download the photos or limit sharing to people who are in the user's 'Circles'. 'Circles' in Google+ are similar to friend lists in Facebook where each category or circle may have different information streams that the users want to share. The 'Circles' can be configured as friends, acquaintances, family and so on. For example, if Alice wanted to share a steganographic image with Bob, Alice can upload the image publicly and Bob will be able to see and download the image from Alice's public profile. On the other hand, Alice can also add Bob to her circles and choose the circle allocated to Bob when uploading the image.

The advantage of disseminating steganographic images in Google+ is that images generated by JP Hide and Seek, S-Tools, StegHide, HIP, GIF-It-Up, F5, SteganPEG, SilentEye and so on, can be directly uploaded with the add photo function in Google+ without any destruction as long as the generated image is in JPEG, BMP, PNG or GIF format and has a resolution of less than 2048 pixel either in height or width. The images will be successfully transported to the intended receiver and the receiver will be able to successfully extract the secret message. As SilentEye generates significant artefacts on its stego-object, using other steganographic tools such as JP Hide and Seek, StegHide, F5 and SteganPEG would be preferable, as these tools are able to generate a steganographic image without perceivable artefacts. Additionally, using JPEG images is less conspicuous as it is a common format for digital photography.

2.5 DIGITAL FORENSICS

Digital forensics first started with computer forensics which mostly dealt with computer related crimes, but, with the prevalence of other digital technologies in our daily lives, activities that we perform via the digital world leave viable digital evidence trails that can aid forensic investigation after a crime or an incident so that an appropriate legal or disciplinary action can be taken accordingly. Nowadays, computer forensics had been extended to include all digital technologies, and is now called digital forensics. Additionally, the concept of computer forensics has also been further divided into the specific areas of mobile

forensics, internet forensics, web forensics, network forensics and lately into the new areas of cloud forensics and social network forensics. What is important in digital forensics is not only to track down the footprint left on digital devices, but to make sure the extracted and analyzed footprint can be allowed as evidence in legal proceedings. The defined digital forensics as

“The use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations.” (Carrier, 2003, para.6)

Throughout the years, many digital forensic investigation processes have been proposed and established but there is only one distinct objective, which is to ensure the process “follows rules that allow the results to be entered into a legal court” (Carrier, 2009, p.26). Thus, for a successful prosecution it is vital to ensure that “when a forensic investigation is launched, it is conducted in a scientific way and with a legal base as support” (Kohn, Eloff, & Olivier, 2006, para.6).

Pollitt (1995) evaluated and mapped the admissible documented evidence in a court of law with the computer forensics process and managed to identify four precedent steps for any evidence that is admissible in a court of law, being: acquisition, identification, evaluation, and admission. The Digital Forensics Research Workshop (DFRWS) recommended identification, preservation, collection, examination, analysis, presentation, and decision as digital forensics processes (Reith Carr & Gunsch., 2002) whereas the National Institute of Standards and Technology (NIST) stated that regardless of the situation, digital forensics investigation should be performed under four basic processes; collection, examination, analysis, and reporting (Kent, Chevalier, Grance, & Dang, 2006). Alharbi, Weber-Jahnke, and Traore (2011) in their research compiled all the processes used in digital forensic investigations and found 18 different processes from a minimum of three processes up to 17 processes. Some of these processes are proposed according to different technology platforms or events, but most of the processes manage to cover the five processes, which are identification, preservation, collection, analysis, and reporting.

The identification process which locates all possible digital evidence, knowing how and where the digital data is stored and justifying the use of tools and techniques needed to acquire the digital evidence (McKemmish, 1999). Digital evidence is fragile. It can easily be tampered with or altered. Therefore, during digital forensic investigation processes, preservation of digital evidence is crucial, the integrity of evidence has to be maintained throughout the entire investigation process until reporting. Any mishandling of evidence during the forensics processes will invalidate the evidence and therefore it may not be admissible in court. This is especially crucial in the collection process, where the digital evidence has to be acquired without making any changes to original data. Usually, this can be accomplished using write-block software or hardware, to ensure data integrity. Once all possible digital evidence has been successfully collected, further analysis can be conducted. The analysis process involves the use of appropriate tools to extract, process, and interpret digital evidence so that it is useful information in relation to the objective of the investigation (Kent et al., 2006; McKemmish, 1999). Finally with all reconstructed evidence and findings a “complete, accurate, and comprehensive” report is presented to the court, which includes a record of steps taken during the analysis (NIJ, 2004, p.19).

The above-mentioned processes are important to determine the reliability of digital evidence for prosecution as a whole. Although the collection and analysis methodologies or procedures could vary in regard to the environment and devices, digital forensic processes of identification, preservation, collection, analysis, and reporting have to be strictly followed. The following section will look specifically at the recommended best practices, methods, and techniques used to conduct social network forensics, web browser forensics, and steganalysis. The stated guidelines will assist in assuring the best practice for conducting a digital forensic investigation associated with steganography on social networks.

2.5.1 Social Network Forensics

Recently evidence gathered from OSNs has been used successfully to testify in court. “Social networks continue to replace traditional means of digital storage, sharing, and communication, collecting this type of data is also fundamental to the area of digital forensics” (Huber et al., 2011). According to a recent survey conducted by Patzakakis (2012), through an online legal database search, from 2010

until March 2012 there were 689 state and federal court decisions across the United States where social media evidence played a significant role. Therefore, a forensically sound method of extracting and analysing data from OSNs is critical.

Mulazzani, Huber, and Weippl (2012) conducted research to identify important data sources that can be extracted from OSNs for forensic investigation analysis without having to have the collaboration of the OSN provider. The authors emphasized that even though the investigator can request relevant data from a service provider; this information may or may not be complete, as the evidence may lack authentication, integrity, and reliability making it unacceptable in a court of law. During data acquisition, the authors mentioned that traditional forensics methods can be used to extract artefacts from local web browser cache files, but they also argued that sometimes information stored in a browser cache is not persistent and therefore not all data is cached. It is also possible to collect data on the network communication layer, which can range from passive sniffing on the network to active attacks similar to sniffing on unencrypted Wifis or it can be in combination with ARP spoofing on LANs. Crawling is also possible but not recommended by the authors as metadata and accurate timestamps do not show up with this method, thus, it is not forensically sound. Metadata and timestamps are especially important in digital forensics as failure to collect and preserve all key metadata from social media may mean a significant risk of having the evidence rejected by the court (Patzakis, 2011).

If the investigation had a court order for interception, then passive logging on the communication layer is possible, but there is a limitation to this approach as collecting information is time-consuming, and the possibility of collecting all the data is difficult according to Mulazzani et al (2012). The authors identified data sources that could lead to viable evidence during a forensic examination on an OSN such as:

- social footprint which is the user's social network's friend connections
- communication pattern – the way in which the user communicates and with whom
- pictures and videos – what was uploaded and who was tagged
- time of activity – the user log in time and when such activity took place

- Apps – what apps have been used, why, and what can be inferred from the social context

Although these are generic data that can help in a forensic investigation, Mulazzani et al. (2012) indicated that it cannot be found on a suspect's hard drive as the information is stored only by the social network provider.

In addition, Facebook, Google+ and other major social network platforms have built in web-based instant messenger features that enable users to communicate with each other instantly through typed messages. "These instant messages can be of great importance to the digital forensic examiner as they can be of great evidentiary value" (Mutawa, Awadhi, Baggili, & Marrington, 2011, p.771). While Yahoo Messenger or MSN Messenger instant messenger applications store conversation as a log file on the user's hard drive, Facebook Chat or Google+ Chat web-based instant chat messengers do not store chatting content on the hard drive. Most often these text-based conversations are stored in RAM only, thus making the chat recovery task difficult with only recent pieces of conversation being able to be restored (Mutawa et al., 2011). Fortunately, since it is web-based, most chat artefacts can still be restored from the web browser cache stored on the hard disk. However, the storage location of the messages can vary according to the browser type. Most chat artefacts from Internet Explorer can be found in \$MFT, Temporary Internet File, \$LogFile, or unallocated clusters whereas in Firefox and Chrome, chat artefacts can be found in _CACHE_001_ and data_1 respectively. Chat artefacts can also be found in pagefile.sys and unallocated clusters regardless of browser type. Most importantly, metadata acceptable to the court as evidence such as the unique message ID, the sender name and profile number, the recipient name and profile number, and the date and time in regard to the message can be clearly extracted (Mutawa et al., 2011).

As mentioned early, none of the data on the OSN is actually stored on the user's computer hard drive as it is web-based content generated by users, therefore web forensics plays a significant role in identification, collection, and analysis on OSN. The above generic data and chat history mentioned by Mulazzani et al. (2012) and Mutawa et al. (2011) can be generalized as a subset of web artefacts and these key elements are the probative evidence that an investigator can look for when conducting an OSN investigation. According to previous Facebook forensics investigation conducted by Wong, Lai, Yeung, Lee,

and Chan (2011), comment, event and chat footprints can be extracted which include the user profile ID, the message contents and corresponding timestamps. These artefacts are usually found in the web browser cache file and on the RAM. However, an investigator has to be aware that not all collection and analysis processes on RAM are feasible as it depends on the power status of the computer. If the computer were already shut down at the time of the collection process, then live RAM acquisition is not possible as rebooting the computer would change the system data. In this case, the investigator could look into the virtual memory swap file named “pagefile.sys” where data are swapped out of RAM and stored in this file during the system’s normal operation. Nevertheless, this data is volatile and it could be lost during the swapping or could still be in the RAM and not yet swapped out (Mutawa et al., 2011; “Retrieving digital evidence,” 2012). Additionally, the data may not even be stored in the hard drive due to the configuration of the operating system (Microsoft Support, 2010).

2.5.2 Web Forensics

Social network artefacts that can be extracted as mentioned in the previous section are mostly dependent upon the web browser cache file. Social network data is not stored on the hard drive. However, since it has to be accessed through a web browser, activities performed through the browser will create log files and be placed on the hard drive. “Almost every movement a suspect performs while using a web browser leaves a trace on computer, even searching for information using a web browser” (Oh, Lee, & Lee, 2011, p.s62). Therefore, it is necessary to review how web forensics works together with social network forensics.

In web browser forensics, web browsing activities can be found in the browser’s cache, cookies, history, and download list (Oh et al., 2011). After this information has been identified and extracted, web forensic analysis can be done to analyze websites visited, the time when the suspect visited a particular website and how frequently he or she visited the website. Consequently, after the investigator has determined that an OSN was involved, a further examination to look for detailed information in regard to the content of the OSN website and associated activities such as photo uploading or downloading, online chatting, social networking emails can be performed. Analyses specifically looking for social network artefacts are called social network forensics. Therefore, web

browser forensics is an integral part of social network forensics. It is significant that the cache file provides more information than other web browser logs (Jones & Belani, 2010a). This is because cache data includes HTML codes, text, images, XML, Java Script, and other sources of information on a website directly downloaded from the web server. This cached data does not only help to speed up the web browsing process every time the same website is re-visited, it certainly aids and provides valuable evidence for a digital forensic investigation (Jones & Belani, 2010b; Oh, Son, Lee, & Lee, 2012).

Jones and Belani (2010a, 2010b) illustrated where and how to analyze browsing activities using two prominent web browsers, Internet Explorer (IE) and Mozilla Firefox. These browsers stored internet activities differently. Activities performed by IE are usually stored in a file named "*index.dat*". "An *index.dat* file is a binary file that tracks user activities such as files opened in Window's explorer, web pages opened in Internet Explorer, and so on" (Craiger, 2006, p.31). There is more than one *index.dat* file used to track browser activities and the location of *index.dat* files may vary depending on the operating system. The advantage of *index.dat* files is that they mapped the cached web page and its corresponding URLs in a unique file in which the operating system can accurately identify and rebuild the web pages visited. Most of the time, the *index.dat* file in Content.IE5 reveals more comprehensive information than others. Table 2.4 shows the *index.dat* file for Windows 7 (Craiger, 2006; Jones & Belani, 2010a).

In contrast, the Firefox browser stores browsing activities separately. Firefox from Version 3 onward stores its Internet history, bookmarks, form field data and cookies files on various SQLite databases and the content of the web pages are stored separately in the cache folder with a cache map file, three cache block files, and separate cache data files when the cache content or metadata is too large to fit into the three cache block files (Jones & Belani, 2010b). These SQLite files and cache files can be found under the user profile folder that is located in the operating system (Table 2.4). Each of these SQLite files has its own .sqlite extension and captures the data that can help in forensic investigation (Pereira, 2009). For example, if an investigator was informed that steganographic images were used on the suspect's OSN website, during the web browser forensics analysis, the investigator can narrow their search to social network URLs and images downloaded from social network websites on the target's

machine. The downloaded URL source, the destination of the file, the time when it was downloaded, the status of the download, whether it was a completed, paused, or cancelled, and a referrer link that indicated the URL link for the downloads would provide useful information for the investigation (Pereira, 2009).

Table 2.4: Browser Cache and Internet History File Locations for Internet Explorer, Firefox and Google Chrome (Adapted from Craiger, 2006; Jones & Belani, 2010a)

Artefacts	Windows	Location
Internet Explorer (IE) Version 5 and above		
Cache	Win 7	\Users\<user>\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\
History		\Users\<user>\AppData\Local\Microsoft\Windows\History\History.IE5\
Cookies		\Users\<user>\AppData\Roaming\Microsoft\Windows\Cookies\ \Users\<user>\AppData\Roaming\Microsoft\Windows\Cookies\Low\
Firefox 3		
Cache	Win 7	\Users\<user>\AppData\Local\Mozilla\Firefox\Profiles\<random number>.default\cache
History, Cookies, Downloads		\Users\<user>\AppData\Roaming\Mozilla\Firefox\Profiles\<random number>.default
Google Chrome		
Cache	Win 7	\Users\<user>\AppData\Local\Google\Chrome\User Data\Default\Cache\
History, Cookies		\Users\<user>\AppData\Local\Google\Chrome\User Data\Default\

According to Pereira (2009), three main databases in the Firefox browser that have the greatest forensic value are places.sqlite, formhistory.sqlite, and downloads.sqlite. Places.sqlite contains all the user accessed URLs and bookmarked information. Formhistory.sqlite records the values entered by the user in the form field on a web page and downloads.sqlite stores all downloads conducted through the Firefox browser. However, Jones and Belani (2010b) mentioned that Firefox internet history does not automatically associate with locally cached content as IE does in the index.dat file. Firefox internet history only reveals the date and time of a particular browsing activity, but is unable to provide the content of such activities. Subsequently, the reconstruction of the Firefox cache file will be needed to identify the relationship between the history activities and the cache content. The reconstruction process is significant as it is

able to reveal the artefacts needed for further analysis, especially incriminating evidence that resides in the content of the web page.

2.5.3 Steganalysis

The process of identifying and discovering the existence of a hidden message is called steganalysis (Ashok et al., 2010; Das, Das, Bandyopadhyay, & Sanyal, 2011; Ibrahim, 2007). The goal of steganalysis is to “identify suspected information streams, determine whether or not they have hidden messages encoded into them, and if possible, recover the hidden information” (Kumar & Pooja, 2010, p.21). The first critical step in the process is to identify a suspected stego-object. Once the stego-object is determined, the process of recovering the secret message proceeds (Das et al., 2011). However, recovering a secret message is challenging for the forensic investigator as the procedures for evaluating steganography can be complex, time-consuming, and sometimes impossible when dealing with unknown objects, tools or techniques. Nowadays, the steganographic object is not only hard to identify visually; discovering the secret message is even harder when steganography and cryptography are used in combination to protect it (Engle, 2003; Ibrahim, 2007).

“Attacks and analysis on hidden information may take several forms: detecting, extracting, and disabling or destroying hidden information” (Curran & Devitt, 2008, p.35). Ibrahim (2007) argued that even though destruction of the hidden information is part of steganalysis, digital forensics is about extracting rather than destroying information as information that is hidden could be incriminating, for example, child pornography or information exchanged for the purposes of drug trafficking or terrorism. Nevertheless, identification of a stego-object and recovery of the secret message is dependent upon the availability of information during the investigation such as

- When only the steganographic object is available
- When the steganographic algorithm is known and steganographic object is available
- When the steganographic object and the original cover object is available
- When both the steganographic and the cover object are available and the steganographic algorithm is known.” (Ibrahim, 2007, para.13)

The difficulty in most cases during forensic investigation in regard to steganography is that “there are no indicators that suspicious file contains some other content” (Cosic & Baca, 2010, p.87). Presuming the stego-object can be identified in the first place; most of the time an investigator may only have the steganographic object without the known cover-object because visual detection is impossible as this approach is to notice the difference between the cover-object and the stego-object (Ibrahim, 2007; Kumar & Pooja, 2010). Therefore, the critical step for steganalysis is first to identify articles with hidden information before any further extraction can be performed.

Provos and Honeyman (2001) conducted research to determine the existence of steganographic content on the internet after the 911 attack. There were allegations that Al-Queda was using steganography for covert communication. A detection framework using a web crawler and statistical attack was established and performed on two million images downloaded from eBay and USENET. The research indicated the existent of steganographic content, but no hidden messages were successfully extracted. Although, statistical analysis is a popular method in steganography detection, Provos and Honeyman (2001) found that images identified by statistical analysis do not guarantee the discovery of secret messages.

Few explanations have been given in regard to the failure of hidden message extraction in Provos and Honeyman’s (2001) research. First of all, detection was limited to JPEG images downloaded from eBay and USENET, therefore it is possible that the analyzed images were not used for steganographic communication. It is also possible that detection had been performed on the wrong transmitting channel. The other possibility could be that the password used was not susceptible to dictionary attack. Furthermore, images on websites are dynamic; they can be added and removed rapidly. Lastly, the research was aimed only at objects created by JSteg, JP Hide and Seek, Invisible Secret, and Outguess 01.3b, and F5, thus any other method would not be likely to be detected (Curran & Devitt, 2008; Engle, 2003; Ibrahim, 2007).

There are several steganographic detection tools that are available either commercially or as open source. StegDetect, by Provos, is still a popular steganography detection tool (Kessler, 2004a). As mentioned before, it can be used to detect JPEG steganographic images that have used JPHide, Invisible

Secret, and Outguess 01.3b or F5. It is able to indicate which steganographic algorithm was used to embed secret messages in a suspicious file (Figure 2.16).

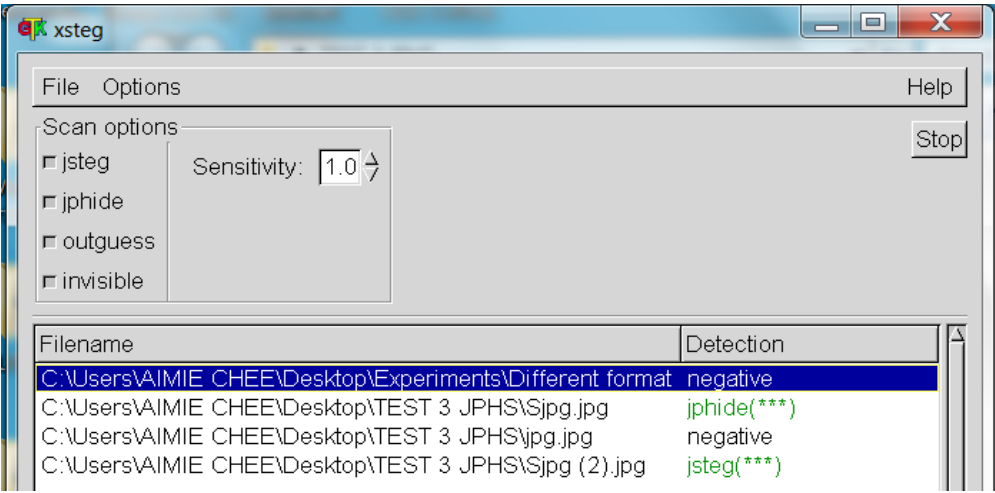


Figure 2.16: StegDetect developed by Neils Provos

Another tool developed by Alfonso Munoz named StegSecret is able to detect images that are embedded with secret messages at the EOF (Munoz, 2007). The prominent commercial forensic tool software AccessData Forensic Toolkit and Guidance Software’s Encase are capable of identifying steganography software in a target’s machine by comparing the data set with a hash set available through HashKeeper, Maresware, and the National Software Reference Library (Kessler, 2004a). StegAlyzerAS and StegAlyserSS developed by Steganography Analysis and Research Center (SARC) are also two common tools used by law enforcement worldwide. StegAlyzerAS is capable of detecting file and registry fingerprints associated with steganographic applications whereas StegAlyserSS is able to detect steganographic files and extract hidden messages (Tone, 2012).

2.6 SUMMARY OF ISSUES AND PROBLEMS

Steganography has always posed a potential threat to information security and digital forensics when it is being misused. Steganography is intended for security purposes in order to achieve confidentiality from adversaries, but ironically, when it is being misused, steganography itself is an adversary to security. Therefore, steganography is good for information protection but it is also a threat to security measures when it is misused. Similarly, steganography is a threat to digital

forensics. As digital forensics seeks to understand who, when, where, what, and how an incident happened, exploitation of steganography has made the investigation work difficult. Incriminating evidence that has utilized steganography to hide its existence will not be revealed unless it is being looking for, and even if it is, identifying innocuous-looking objects that have information embedded in them can be impracticable if the investigator lacks knowledge and is unable to find an effective guideline that provides a systematic approach to steganography-related investigation.

Traditionally, investigating steganography was searching for steganographic tools installed in the system and using the tool to lead the investigator to the type of carrier that might have been used for steganography. However, due to technological advancement in removable storage, some steganography tools can now be executed through portable hard drives or USB flash drives without them being installed on the computer system. Additionally, websites have been used as a popular platform to propagate steganographic images, and as the sheer amount of data transmission on the Internet is so vast, it is unfeasible for law enforcement to screen through all types of digital media to look for steganographic content, thus illicit traffic will normally slip through undetected. Moreover, the presence of OSN websites may make screening unachievable as OSNs have privacy configurations so that only permitted users can see each other's content. Furthermore, the dynamic nature of an OSN website is also a challenge as posts can be deleted very easily.

With the presence of OSNs, secret messages can be disseminated into a few segments and embedded into more than just photos to be uploaded onto an OSN without generating any attention as photo sharing is a common activity on OSNs and, with the new Facebook file sharing feature, steganography propagation is even more streamlined. Furthermore, steganography algorithms and steganography tools are easy to obtain on the Internet, either freely or commercially, and this has also increased the use of steganography. An intelligent criminal may manipulate the available algorithm and develop their own tool for steganography, whereas an average criminal may just download the free tool for criminal purposes. The use of steganography on social networks is indeed possible. It can be propagated with different steganographic tools and different features of OSNs without a need for sophisticated computer skills. Therefore, there is a need

to be prepared for systematic digital forensics examination in relation to steganographic activities on OSNs.

Guidelines have been developed and suggested by various researchers in conducting OSN and web browser forensics, but none of them were concerned with steganographic content. Most OSN forensics are focused on how and where to look for text based artefacts in chat, post, comment and message features, but do not mention downloaded and uploaded photo artefacts, not to mention steganography. Web browser forensics have focused on web page reconstruction, web caches and histories to determine which web pages the target has visited; the pictures in the reconstructed web pages will be seen as they are, without further evaluation with the possibility of steganographic content.

Evidently, there is a lack of routine examination on steganographic content when conducting digital forensics investigation related to web sites forensics especially OSNs. It has been proven that information in OSN content can aid digital forensics investigation; however with the aid of steganography, the incriminating evidence may circumvent detection. Therefore, there is a need to include steganographic evaluation in digital forensics investigation and a need for a guideline on how to conduct digital forensics examination regarding steganographic content in OSNs so that digital forensics investigators are prepared for such a situation when it is occurs.

2.7 CONCLUSION

Chapter 2 has reviewed comprehensive literature ranging from the state of the art of steganography to its impact on digital forensic investigation. The chapter started with an overview of classical steganography and modern steganography and then further identified differences between steganographic classifications and how they can be utilized on different sources. Digital image formats were reviewed in order to provide a better understanding of how digital images represent colour as this fundamental knowledge can aid understanding as to how the bits and bytes of colours in an image can be manipulated for secret message hiding. The literature review continued with possible image steganographic techniques that can be applied to OSNs, how OSNs process images before

uploading them, OSN photo sharing capabilities, and lastly reviewed digital forensics investigation associated with OSNs, web browsers, and also steganalysis.

Problems and issues caused by the misuse of steganography as discussed in Section 2.6 have highlighted a need for further research in evaluating steganographic content related to OSNs, especially digital images, during digital forensics examination. Digital forensic investigators have to be well prepared and know where to extract and how to examine steganographic artefacts that are left behind on a computer system in order to reveal hidden incriminating digital evidence. Any improper handling may destroy the hidden evidence and hence affect the findings. Likewise, improper forensic handling may affect its admissibility in a court of law. Therefore, the focus of the proposed research is to find a forensic ally sound and efficient way of examining steganographic content on OSNs while also determining the necessity of including steganographic evaluation as a routine check when conducting digital forensic examinations specifically on OSNs.

Chapter 3 is to formulate the research design by reviewing other similar works related to the research area and to establish the main research question derived from Section 2.6. Accordingly, the associated hypothesis and sub questions will also be determined. Lastly, the limitation of the research design will be identified and discussed at the end of Chapter 3.

Chapter 3

Research Methodology

3.0 INTRODUCTION

Chapter 2 has reviewed the literature that is relevant to the topic area ranging from steganographic techniques, image formats, to forensic investigation that is associated with online social networks and web browsers. Subsequently, the problems and issues in the research area have also been identified. The purpose of Chapter 3 is to construct an appropriate research methodology that suits the problem and research questions that are to be derived from this area.

Five similar studies are analyzed and studied in Section 3.1 in order to learn from previous researchers and to develop a research methodology that is suitable for the context of the proposed research. In order to shape the research design, in Section 3.2, selected issues from Section 2.6 and the five similar studies in section 3.1 are reviewed to identify a researchable problem and to formulate relevant questions. Following that, the research sub-questions and related hypotheses to be tested are developed in Section 3.2.3. Research phases are adopted from the empirical approach and are described in Section 3.2.4 with a process diagram. A data map is constructed in Section 3.2.5 to represent and communicate the relationship between the research phases and the research sub-questions, the tested hypotheses and the main research question.

Section 3.3 defines the data requirements for the proposed research, which consists of investigative case scenarios, data collection, data processing, data analysis and lastly data presentation. This section is crucial as it enables the researcher to plan thoroughly and identify the necessary data required for the research so that the research evaluation can be performed accordingly. Finally Section 3.4 discusses the limitations of the proposed research methodology.

3.1 REVIEW OF SIMILAR RESEARCH

Five similar works are studied and analyzed in order to learn from others of how to develop an appropriate methodology for the proposed research. Previous literature reviewed in Section 2.5 has given some insights regarding where to look for potential sources of evidence when conducting web browser forensics and online social network forensics. The following five relevant works aim to provide similarity to the research area and help to derive a methodology that can be adopted in conducting forensic investigation of steganographic activities specifically on images that are found on online social networks (OSNs).

Sections 3.1.1 and 3.1.2 show how previous researchers searched for steganographic content in images. Berg et al. (2003) employed a machine learning analysis to detect hidden messages in the steganographic images (Section 3.1.1) whereas Provos and Honeyman (2001) conducted a statistical analysis to determine the signature of each steganographic tool and developed a steganography detection application based on statistical analysis (Section 3.1.2). Section 3.1.3 is the research conducted by Zax and Adelstein (2009), which emphasized identifying and detecting steganographic tools' artefacts as part of the initial forensic investigation. Although Section 3.1.4 focuses on finding hidden data in the NTFS disk image, the empirical methodology employed in the research was found to be a relevant approach to the realm of the research design. The review conducted in Section 3.1.5 provides a set of reliable digital forensic investigation steps useful for the proposed research.

3.1.1 Searching For Hidden Messages

Berg, Davidson, Duan, and Paul (2003) in the article - *Searching for hidden messages: Automatic detection of steganography*, conducted research on steganography detection using a machine learning (ML) approach. The finding of the research showed that the ML algorithm was able to successfully differentiate a clean object from a steganographic object by identifying the unique feature of "the available space within the file to hide a message", which is called as a canvas (Berg et al., 2003, p.51). The results of the research reported that the ML techniques are not only capable of detecting secret messages embedded in both

lossy and lossless image formats, but also contributed a general framework for steganalysis on multiple media and a variety of content.

The authors argued that the common manual steganography detection that uses statistical tests to identify the unique signature of a steganographic technique or a clean file type is not an effective approach as it has a high false positive rate and that it is not feasible to identify the signature of steganographic techniques that preserve the statistical properties of the cover-object when embedding the secret message.

Berg et al. (2003, p.53) used an experimental methodology in their research with the assertion of “automated learning and data mining techniques can potentially create models that successfully attack a variety of steganography techniques, including previously unseen variations of existing techniques”. In order to accomplish the objective of the research, the researchers had to test the data mining and machine learning capability to identify hidden messages of a specific steganographic technique. Three popular machine learning algorithms, decision tree, error back-propagation artificial neural networks and naïve Bayes classifier were chosen for testing on hidden messages embedded in both JPEG and GIF format images.

In the first phase, JSteg Version 4, a JPEG steganographic tool was tested. For data collection, 50 natural images for each dataset of flowers, mountains and trees were generated for the machine learning algorithm to learn the difference between clean images and images with hidden secret message. The datasets consisted of half clean images and half steganographic images. The features of the images were also calculated. “Each image is represented by the unconditional entropy, positional conditional entropy values, and transition probabilities of the DCT coefficient’s LSB” (Berg et al., 2003, p.53). The unique features included the mean entropy for the entire image, the mean and standard deviation of entropy across each block in the image, the mean and standard deviation across each block of the transition probabilities and so on. Altogether there were three datasets and each dataset consisted of 50 instances (images) and 51 features.

After the datasets were created, each of the machine learning algorithms was executed on each dataset using supervised five-fold cross-validation. The results from machine learning experiment showed that the error back propagation artificial neural network algorithm out performed decision tree and naïve Bayes

algorithm in detecting embedded secret messages especially on the flower and mountain datasets. A comparative analysis was then conducted on both machine learning techniques and statistical attack techniques. Statistical attack was performed using a steganalysis tool called StegDetect. The experiment's results showed that the data mining evaluation outperformed StegDetect in one of the datasets and, evidently, the error back propagation artificial neural network machine learning technique showed a higher accuracy in detecting the existence of secret messages in all three of the datasets than did StegDetect.

A similar experiment was set up and conducted on steganographic images created using GIFShuffle. However, the selected features of the experiment on GIF images were different from JPEG. The conducted test used “unconditional and conditional entropies of indices to represent each GIF”. After comparative analysis, the experimental results showed that all three algorithms were capable of detecting secret messages embedded using GIFShuffle with neural network the best performer with more than 85% accuracy in a supervised data mining.

Although the data mining and machine learning techniques used in the experiment proved successful in the detection of JSteg and GIFShuffle, it was weak in detecting steganographic technique that used the F5 algorithm and JPHide and Seek algorithm due to the specific features of the steganographic algorithm in the compressed image format and “an analogous situation for GIF format” (Berg et al., 2003, p.54).

3.1.2 Detecting Steganographic Content on the Internet

The allegation that terrorists used image steganography for covert communication on the Internet for the September 11 terrorist attack in the United States motivated Provos and Honeyman (2001) to conduct research to find out whether steganographic content exists on the Internet to ascertain the legitimacy of the claim. Provos and Honeyman (2001, para.2) established “a detection framework that includes tools to retrieve images from the World Wide Web and automatically detect whether they might contain steganographic content”. A web crawler was used in the detection framework to download JPEG images from suspected websites and statistical analysis was performed on the downloaded images to identify steganographic images. Statistical analysis is only capable of identifying the possibility of a hidden message, but is unable to retrieve the

content of the hidden message. Therefore, Provos and Honeyman (2001) established a distributed computing framework that uses dictionary attack in order to recover hidden messages.

The statistical analysis performed in the detection framework was using mathematical calculation on the image's statistical properties discovered deviations from a norm to distinguish clean images from steganographic images. The authors measured the entropy of the redundant data and predicted that images with embedded secret message would have higher entropy. χ^2 statistical tests were used on steganographic images that created using JSteg, JSteg-Shell, JP Hide and Seek, and OutGuess to “determine whether an image shows distortion from embedding hidden data” by calculating the “probability of embedding for different parts of an image” (Provost & Honeyman, 2001, para.27 & 28). The test proved that each of the steganographic tools has its own unique distortion characteristic on the steganographic images. Hence, these characteristics, also called signature, can be used by the automated detection framework to determine which steganographic tools have been used in a particular steganographic image. This detection framework was implemented by Provos and Honeyman (2001) in StegDetect, an automated detection tool for steganographic content in JPEG images. It uses a one to three star rating to indicate the level of confidence in the detection.

Before the tool was used to detect images downloaded from the suspected website, the detection sensitivity of the tool was verified on 1500 images taken on a Fuji MX-1700 digital camera, that were used to generate steganographic images using different steganography tools, JSteg, JPHide 0.5 and OutGuess 0.13b. The test results showed that “the smaller the message, the harder it is to detect by statistical means” (Provost & Honeyman, 2001, para.70). StegDetect showed a convincing result for JSteg detection, however, the tool is unable to detect an embedded secret message that is smaller than 50 bytes, where the false negative rate is at 100%. When the embedded secret message was more than 150 bytes, the false negative rate fell to 10% for JSteg whereas JP Hide and Seek was at least 20% in all cases and OutGuess 0.13b had at high false negative rate of around 60%.

With the known capability of StegDetect after the preliminary test, an experiment was carried out to detect images downloaded from the websites of interest, which were eBay and the USENET archive. A web crawler named Crawl

developed by the authors was used to perform the downloading and was integrated with StegDetect for automated detection. Two million images were downloaded from eBay and the analysis results showed that 17,000 of the images were likely to have steganographic content and 15,000 images were detected which had used JPHide. A further study on an additional one million images downloaded from the USENET archive was conducted and the false positive analysis from both eBay and USENET is shown in Table 3.1.

Table 3.1: Percentage of false positives from images obtained from the Internet
(Provos & Honeyman, 2001, para.79)

Test	False Positives	
	eBay	USENET
JSteg	0.003%	0.007%
JPHide	1%	2.1%
OutGuess 0.13b	0.1%	0.14%

The next phase of the experiment was to verify the identified images had embedded secret message content using the statistical test. According to the authors, the statistical test performed by StegDetect was to indicate that a particular image might be embedded with a secret message by a specific tool, thus it raises an alarm for the investigator. However, it could not guarantee the existence of hidden secret messages. Therefore, StegBreak was created by the authors in order to recover the hidden message. StegBreak used dictionary attack to recover the password that was used to embed the secret message. StegBreak was “running on a large cluster of loosely-coupled workstation for the dictionary attacks” (Provos & Honeyman, 2001, para.114) and the authors assumed that weak passwords were used for the steganographic system. The dictionary of about 850,000 words were used to attack the identified steganographic images from eBay and 1,800,000 words including four-digit number and short pass phrases were used to attack the identified steganographic images from USENET. To ascertain whether StegBreak performed the attack properly, tracer images were inserted into every StegBreak job, and it showed that the dictionary attack correctly found the password for the tracer images. However, the research was not able to recover any genuine hidden message from the suspected websites on the Internet.

3.1.3 Forensic Artefacts of Uninstalled Steganographic Tools

Most of the time to embed a secret message in an image requires some kind of steganographic tool, as such there must be traces left behind by the tool in directories or registry keys even if the program has been removed or uninstalled. Therefore, the main purpose of the research conducted by Zax and Adelstein (2009) was to show an alternative approach in conducting steganography related investigation by performing a quick search for steganography tools that were used on the system instead of detecting steganographic content. The authors conducted an experiment to identify the “traces that left behind after a number of freely available steganography tools were installed, run, and uninstalled” (Zax & Adelstein, 2009, p.25).

The authors argued that digital forensic investigators should not overlook the use of steganography in every investigation and it is necessary for investigators to be able to quickly detect the presence of common steganography tools and to determine whether further steganographic analysis is warranted as a detailed steganalysis is time-consuming. The authors suggested performing a quick and “efficient search for steganography tool as part of the initial triage phase” and if traces of a steganography tool are discovered, a further evaluation can be conducted later in the forensic analysis phase of the investigation (Zax & Adelstein, 2009, p.26). The research was specifically to answer the research question: What forensics artefacts remain after steganographic tools have been removed or uninstalled?

An experiment was conducted to answer the research question and to determine the accuracy of the assertion that steganographic tools leave artefacts in the file system and registry. If the assertion is in fact accurate, what artefacts can be identified? The methodology for the experiment was first to select popular Windows-based steganographic tools available for download as freeware from well-known software sites. There were altogether 20 steganographic tools used in the experiment. In the second phase, a controlled environment was set up with a virtual machine (VM) using Virtual Box so that the experimental environment was separate from the actual machine in order to provide a clean system without a need to reformat the operating system. This method also protects the physical

machine from malicious code that might embed in running the steganographic program.

Once the experimental environment was set up, steganography tools were downloaded, installed or unzipped, and executed with sample data, and then uninstalled or deleted if the tool did not require installation. Those without installation were usually packed in a zip file. All processes were monitored by Windows SysInternal tools File Monitor and Registry Monitor. The creation of new files, directories, and register keys performed by the steganographic tools was captured by the monitoring tools and enabled the authors to determine artefacts after deletion or un-installation. The results of the experiment showed that of the 20 tested tools, 8 to 9 tools left obvious and permanent artefacts such as “folders, files, and registry keys bearing the names of the programs or the authors” whereas there was no significant evidence for the other tools tested (Zax & Adelstein, 2009, p.27). Also, the experimental results indicated that steganographic tools that require installation have a tendency to leave a more permanent footprint than those packed in zip files not requiring installation. However, the authors highlighted that even though there are permanent artefacts such as the generated steganographic files, these cannot be presented as evidence using the present forensic method as those files cannot uniquely prove the use of a specific steganography tool.

This forensic method is suggested by the authors as a quick check in the initial phase of investigation to determine whether “steganography tools were at some point used on a computer, even if the tools were later uninstalled or deleted” before searching directly for steganographic content on the suspect’s computer system which requires in-depth, time-intensive analysis (Zax & Adelstein, 2009, p.29). The authors also mentioned other benefits of implementing this method. The initial, quick finding allows the investigator to proceed with some clues and secondly, it minimizes the scope of the search so the investigator can look for specific carrier files type generated by the steganographic tool detected. This is called functional analysis and relational analysis in investigative reconstruction, in functional analysis the investigator will be able to “consider all possible explanations for a given set of circumstances” for example, if a steganographic tool is detected, then the suspect may possibly be involved in evidence hiding (Casey, 2004, p.124). Whereas in relational analysis whether an object or person

was in relation to another object or person is determined. For example, if a specific steganographic tool was detected, then the types of file that could be used as carrier object would be determined (Casey, 2004).

3.1.4 Effective Digital Forensic Analysis of the NTFS Disk Image

This research paper was conducted by Alazab, Venkatraman, and Watters (2009). The purpose of the research was to focus on the analysis phase of the digital forensic investigation process to acquire necessary hidden evidence from a computer system after an intrusion. The authors argued that “many current forensic techniques have failed to identify malicious code in hidden data of the NTFS disk image” (Alazab et al., 2009, p.551). Therefore, their research study was to tackle this problem by conducting an empirical study to investigate the effective techniques which analyze and acquire hidden evidence from the NTFS disk image.

The experimental method was used in their empirical study. In the first phase, digital forensic tools that covered a comprehensive set of functionalities were carefully selected. Dd or dcfldd V1.3.4-1 disk imaging tool for sector-by-sector imaging; Hexedit, Frhed1.4.0, and Strings V2.41 utilities tools were selected for evidence collection for binary code reading; The Sleuth KIT (TSK) 3.01 and Autopsy NTFS disk analysis software; NTFSINFO v1.0 forensic analysis tools were selected for exploring and extracting intruding data and hidden data. The purpose of the experiment was not only to investigate an effective analysis technique but also to test the effectiveness of the selected tools in the first phase.

Then, test data were created on a Pentium ® Core™ 2 Duo CPU, 2.19GHz, 2.98 RAM operated with Windows XP Professional NTFS file system. A three-stage forensic analysis was proposed by the authors for the experiment. Stage 1 was called hard disk data acquisition. In this stage, dcfldd and dd disk imaging utilities were used to acquire the NTFS disk image from the hard drive that consisted of the test data and verify the message digest 5 (MD5) hash values to ensure data integrity. Stage 2 was evidence searching, where evidence related to the misuse of the system was searched for. Three tools, string command, Frhed hexeditor, and WinHex, were used for keyword or phrase searching. Stage 3 involved analysis of the information extracted from the NTFS file system “that contributed towards meaningful conclusions of the forensic investigation” (Alazab

et al., 2009, p.552). This three-stage forensic analysis, covered nine steps of forensic investigation as shown in Figure 3.1, which were: 1) policy and procedure development 2) hard disk acquisition 3) checking of data integrity 4) extraction of MFT in the boot sector 5) extraction of \$Boot file and backup boot sector, 6) comparison of boot sector and backup boot sector, 7) checking of data integrity for the image boot and backup boot sector, 8) extraction of the ASCII and UNICODE and 9) documentation and reporting.

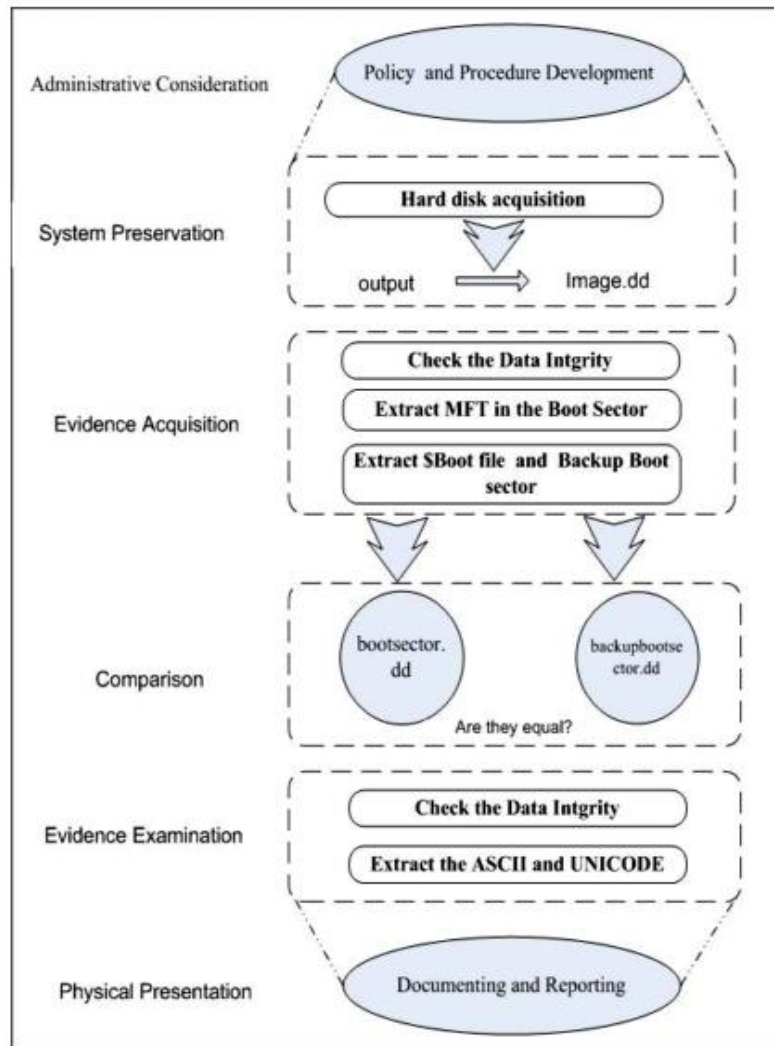


Figure 3.1: Forensic investigation steps (Alazab et al., 2009, p.553)

According to the authors, boot sector analysis that followed Step 4, using WinHex hexeditor and NTFSINO enabled them to extract useful information such as “the size of clusters, sector numbers in the file system, starting cluster address of the MFT, the size of each MFT entry, and the serial number given for the file system” (Alazab et al., 2009, p.554). However, the experiment also revealed that tools

used in the boot sector analysis were unable to detect certain hidden data in the boot sector.

In order to reveal the hidden data, a manual analysis of the \$Boot data structure of the NTFS file system was performed by comparing the MD5 hash value of both the boot sector and the backup boot sector. This technique can clearly identify whether the inspected NTFS file system contains any hidden data because without hidden data, both sectors would have the same MD5 hash values; if not, then hidden data is in the sector. Through this empirical study the authors have presented some effective search techniques that could successfully identify malicious hidden data in \$Boot files and also uncovered the weaknesses of current forensic software which is not able comprehensively to identify hidden data in the boot sectors.

3.1.5 Computer Forensics Guidance Model with Cases Study

In this research paper, Noureldin, Hashem, and Abdalla (2011) conducted a systematic analysis research based on their previously published works on *“Digital forensics model and computer forensic teams responsibilities and process”* (Noureldin et al., 2011, p.564). According to the authors, this research refined and constructed a more comprehensive model with step-by-step investigative processes. Two real world case studies that came with different scenarios, platforms and environments were used to validate their proposed computer forensic guidance model. The research results showed that the deployed model fit well for computer crimes and intellectual property right (IPR) crimes. On the other hand, the model was also capable of handling investigative cases that involved secret or hidden data stored in hidden areas such as Host Protected Areas (HPA) and Device Configuration Overlays (DCO).

The previous works were refined in this research using flow charts to aim for better visualisation of the sequence of investigative processes so that each step is clear and easy to follow. The model was “structured to encourage a complete, rigorous investigation, ensure proper evidence handling, and reduce the chance of mistakes created by preconceived theories and other potential pitfalls” (Noureldin et al., 2011, p.564). The procedures in each phase were also illustrated in detail so that the model can easily be adopted by the investigator when conducting an

investigation. The model's phases used by the authors in their case studies are shown in Figure 3.2.

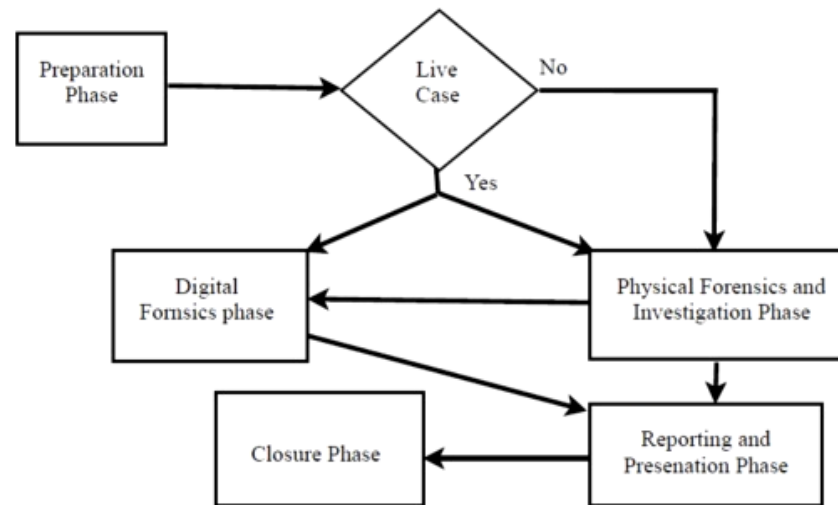


Figure 3.2: Model Phases (Noureldin et al., 2011, p.564)

Their research model consists of five phases: preparation, physical forensics and investigation, digital forensics, reporting and presentation, and closure.

The digital forensics phase is the one that will be focused in this review. In this phase, the computer is assumed to be the secondary crime scene and the objective is to identify, collect, and analyse the artefacts that answer the questions of who, what, where, when, why, that map the evidence found in the physical crime scene. The authors suggested the steps in Figure 3.3 to answer those six questions.

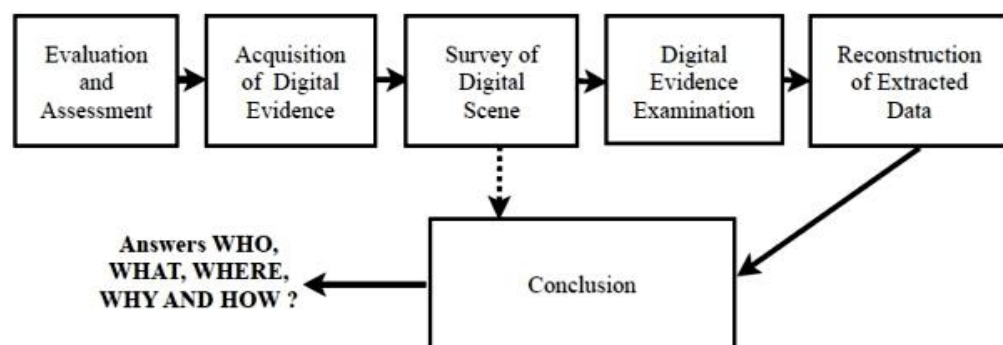


Figure 3.3: Digital Forensics Phase (Noureldin et al., 2011, p.566)

Step 1 is evaluation and assessment, in this step the investigator has to determine the condition of the physical evidence. Activities here are proper chain of custody documentation, determining the necessary tools, and so on before starting digital investigation procedures. Step 2 is acquisition of digital evidence where an exact

copy of the target machine is imaged. It can be a live acquisition or a power-off acquisition or both, depending on the case. A flow chart of evidence acquisition is depicted in Figure 3.4 by the authors. Step 3 is survey the digital scene, this phase is similar to a preview, where the location of significant evidence is identified and the suspect's level of technical competency is evaluated so that the investigator can determine the necessary investigative techniques or approaches to search for additional evidence. Step 4 is digital evidence examination. This step is to locate, extract, identify, and possibly uncover all the probative data that can be used to analyse and reconstruct the crime scene. The authors highlighted that it is necessary to extract deleted, hidden, camouflaged, or unavailable-to-view data prior to the full analysis. After all evidence is gathered and extracted, this is where Step number five, reconstruction of extracted data came in. Several analyses can be performed during reconstruction depending on the nature of the case, ranging from timeframe analysis, data hiding analysis, application and file analysis, ownership and possession analysis, log file analysis, email message analysis and network analysis. The reconstruction will help "to produce a clear picture of the crime and identify the missing links in the picture" (Noureldin et al., 2011, p.567). Finally, to conclude the case, findings collected from both the physical and digital forensics phases have to be considered in order to determine who was involved in the digital events.

In section three of the article, two real world cases were studied and investigated using the suggested model. One was a case that related to national security where the hard drive was suspected to contain national security information. Case number two involved examination of a suspect's machine which had a high possibility of containing pirated software. Both case studies have proved that the model "is general with respect to technology as well as abstract enough that it can be applied to law enforcement investigation and corporate investigation" as both case studies covered diverse scenarios, platforms and environments (Noureldin et al., 2011, p.571).

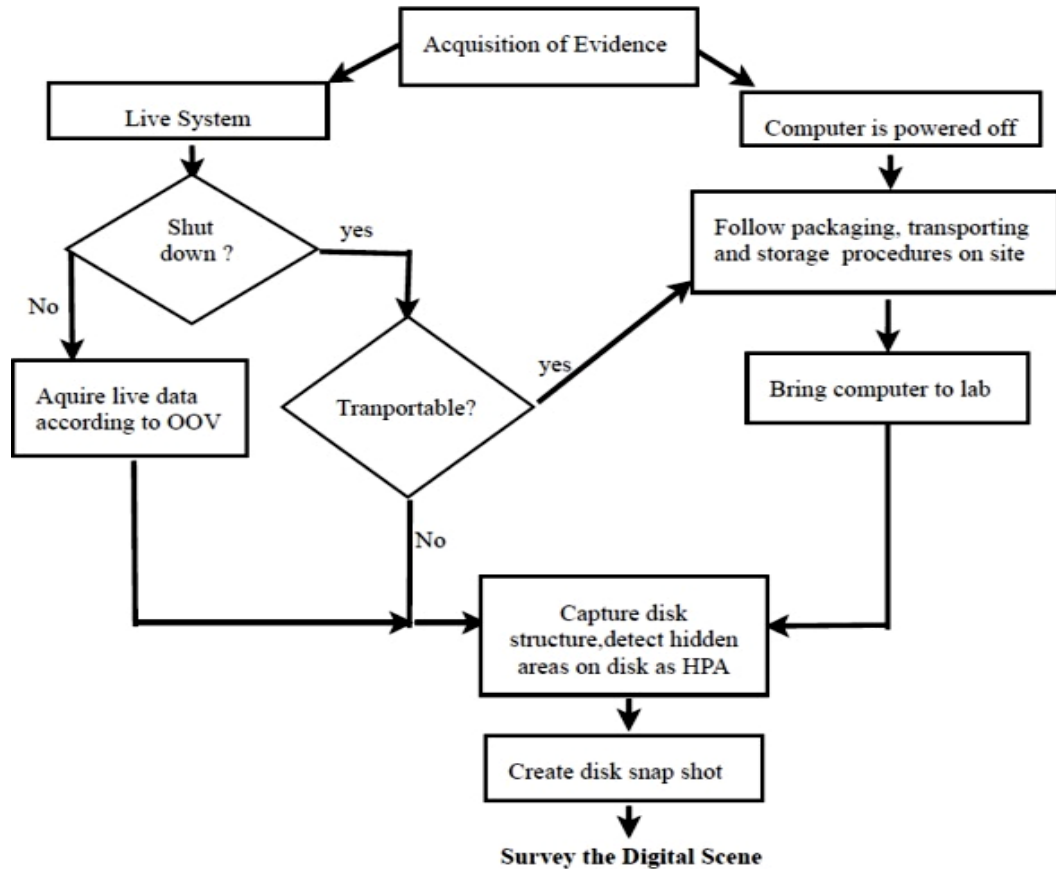


Figure 3.4: Digital Evidence Acquisition (Noureldin et al., 2011, p.566)

3.2 RESEARCH DESIGN

Five similar research projects have been analyzed and identified in Section 3.1 to develop an effective research methodology that can be adopted for the proposed study. An empirical study will be conducted for the proposed research with a systematic approach to investigate the identified problem areas. The American Psychological Association defined an empirical study as “study based on facts, systematic observation, or experiment, rather than theory or general philosophical principle” (“American Psychological Association,” 2012). The most interesting part of an empirical study is that it helps a researcher “to build upon what is already known” (Hani, 2009, para.6). In the proposed research, various steganographic techniques using available steganographic tools will be experimented on in online social networks (OSNs) in order to identify OSNs’ capability and limitation in regard to steganography. Later, a systematic forensic examination using some current popular forensic tools will be conducted on a case scenario to learn what artefacts can be found in a system after steganographic

objects have been posted on an OSN. The results of the experiment will be used to answer the research question and the investigative steps conducted in the experiment will be used as a guideline for systematic steganographic evaluation.

A discussion of the five published studies in Section 3.1 will be reviewed in Section 3.2.1. A problem derived from Section 2.6 in the area of forensic investigation of steganographic images propagated in social media will also be discussed in Section 3.2.2. The use of steganography is an issue to forensic investigators because it will not be discovered unless it is being looked for. It is a type of anti-forensics method that effectively makes the criminal evidence invisible from plain sight. Therefore, an evaluation of steganography has to be included in the routine check when conducting a forensic investigation as hidden information could be the source that helps to reconstruct a crime scene or it could be the probative evidence itself.

After the review of similar studies and problem areas in Section 3.2.1 and 3.2.2 respectively, the research question will be derived in Section 3.2.3 including related sub questions. Following that, the research hypotheses will be established. Section 3.2.4 will present the research phases (Figure 3.5) that include: Phase 1: Pre-test and evaluate OSN capability, Phase 2: Steganographic exploitation on an OSN based on case scenarios, Phase 3: Acquire and extract evidence, Phase 4: Comparative analysis with known artefacts, Phase 5: Method recommendation. Lastly, Section 3.2.5 presents the proposed research data map (Figure 3.6) that logically bonds each stage of the research process to the relevant research question, sub-questions, and hypotheses accordingly.

3.2.1 Summary of Related Studies

The five related studies have been reviewed in Section 3.1 and what was learnt from these studies is summarised to provide guidance for research in this area. The first and second studies by Berg et al. (2003) and Provos and Honeyman (2001) focused on how to identify and detect images that are embedded with secret messages. Both studies used different approaches. Berg et al. (2003) used a machine learning approach to compare clean and steganographic images whereas Provos and Honeyman (2001) used statistical tests to calculate the distortion probability in clean and steganographic images. Provos and Honeyman (2001) found that each tool has its own distinct distortion probability and therefore the

pattern was used to develop an automated detection tool that can identify unique signatures in a steganographic image.

Although Berg et al. (2003) had a different research direction from that which is intended in this research, their approach to the research area can be adopted. Berg et al. (2003) conducted an experiment to test whether machine learning can identify steganographic images created by a specific steganographic tool. Similarly, in Phase 1 of the proposed research an experiment will be conducted to test whether steganographic techniques can be applied on an OSN. Provos and Honeyman's (2001) research studies helped them to develop a steganography detection tool, StegDetect that is capable of identifying steganographic images. In the proposed research a commercial steganography detection tool called StegAlyzer will be used and StegDetect will be adopted for a comparative analysis.

The study conducted by Zax and Adelstein (2009) focused on identifying artefacts that are left behind by steganographic tools in a system registry. Their research and the proposed research have a similar perspective, which is that a “digital forensic investigator cannot simply ignore steganography” (Zax & Adelstein, 2009, p.25). Zax and Adelstein (2009) stressed identifying the steganographic tool quickly especially in the early triage phase rather than searching data files that may contain steganographic content using steganalysis. Zax and Adelstein (2009) argued that it is inefficient to have steganalysis as a general practice as it is too time consuming. However, the proposed research has a different point of view, although an initial search for a steganographic tool is necessary; it cannot guarantee that a system does not contain steganographic content if a steganography tool cannot be identified in the triage examination as some steganographic techniques do not leave footprints after execution. Early detection of a steganography tool in the initial triage phase as suggested by Zax and Adelstein (2009) is important to adopt, however, detection of steganographic content and steganalysis are also necessary as the hidden message could contain crucial information for the investigation.

Alazab et al. (2009) conducted an empirical study to investigate digital forensic techniques that could be used to analyse and acquire evidence from a NTFS system that had been hidden. Although the research was more aimed at disk space steganography, the empirical study and data collection methods in the study

can be adopted for the proposed research. Lastly, the proposed forensic guidance model by Nouredin et al. (2011), which had been proven successful in two real world case studies, will be adopted for the proposed research to ensure that the digital forensic investigation proceeds in a forensically sound manner.

3.2.2 Review of the Problem Areas

In Chapter 2, Section 2.6 the threats and challenges that steganography poses to security personnel and digital forensic teams have been reviewed. As discussed in Section 2.5.3, digital forensic investigation of steganography has always been considered a complex and time consuming task. Most of the time steganography evaluation is not included in the general forensic practice during a forensic examination, yet, this process is imperative. If the probative evidence is concealed with steganography, the investigator will not be able to find what they are looking for as the point of steganography is to make the information unperceivable. It is similar to the physical world where a killer tries to bury his or her weapon or the corpse itself under the ground, his or her intention being to conceal the probative evidence from law enforcement. So, in the digital world, steganography is capable of making the incriminating information invisible to the investigation or forensic processes.

One of the major reasons why steganographic evaluation is not included in a routine check is because steganographic examination is believed to be complicated and time consuming (Hosmer & Hyde, 2003; Sheetz, 2003; Zax & Adelstein, 2009). Therefore the proposed research is not only to highlight how steganography can be exploited on an OSN, but also to find an effective and systematic approach to tackle this issue using the right tools for steganographic evaluation and possibly minimizing evaluation time.

As reported in Section 2.6, a current web browser forensic examination and social network forensic examination does not mention about steganographic evaluation and how to examine the evidence related to steganographic content. Thus, understanding how steganographic activities can be performed on an OSN is useful as it will make investigators aware of the techniques and they will know exactly what to look and how to look for hidden evidence on an OSN. As Michael Sheetz wrote in the article, *Reading between the lines: Steganography*, “It is imperative that you approach every investigation with the assumption that the

suspect could benefit from steganography in some way” (Sheetz, 2003, p.49). Hence, when human knowledge is integrated with a systematic approach, an effective investigation method that deals with steganography can be established.

3.2.3 The Research Questions & Hypotheses

The research question is derived from the literature review conducted in Chapter 2. Various steganographic techniques, specifically image steganography, have been discussed in the literature review in Section 2.3. The strengths and weaknesses of the embedding techniques were also highlighted. Section 2.4 shows how OSN features assist steganographic activity. Although, steganography is a real threat to forensic investigation, the issue is not actively addressed in forensic examination and is being neglected due to the complexity of investigation. Therefore, the main research question for this proposed research is stated as:

Should digital forensic investigators include steganography as a routine check in their standard digital forensic investigation procedures in relation to online social networks?

Following on from the proposed research question and the problems that have been discussed in Section 3.2.2, the research hypothesis is asserted as:

That digital forensics investigator should include steganographic evaluation as a routine check in their standard digital forensic investigative procedures in relation to online social networks as the footprints of steganographic tool, its usage, and the steganographic image can be identified.

In order to answer the research question and evaluate the research hypothesis, sub-questions have been derived which can be answered accordingly:

Sub-question 1 (SQ1):

Can the automated steganalysis tool StegAlyzerAS identify steganographic tool artefacts in the target’s system?

Sub-question 2 (SQ2):

Where are identified steganographic tool artefacts located?

Sub-question 3 (SQ3):

How long does it take StegAlyzerAS to identify steganographic tools’ artefacts?

Sub-question 4 (SQ4):

Can StegAlyzerSS identify the uploaded and downloaded steganographic images from an OSN?

Sub-question 5 (SQ5):

Where are the identified steganographic images located in the target system?

Sub-question 6 (SQ6):

Is the process of determining steganographic images tell from which OSN these images were downloaded or uploaded?

Sub-question 7 (SQ7):

How long does StegAlyzerSS take to identify steganographic images?

Sub-question 8 (SQ8):

Can StegAlyzerSS extract the secret message embedded in the images?

Sub-question 9 (SQ9):

How long does it take StegAlyzerSS to extract the secret message?

From the research sub questions, hypotheses are established as follow:

Hypothesis 1 (H1):

When conducting a digital forensic examination, the footprint of a steganographic tool or its usage can be identified.

Hypothesis 2 (H2):

When conducting a digital forensic examination, the steganographic images can be identified.

Hypothesis 3 (H3):

The hidden data in identified steganographic images can be extracted when conducting a digital forensic examination.

3.2.4 Research Phases

Based on an empirical approach, the proposed research is divided into four phases as shown in Figure 3.5. Phase 1 is a preliminary test to experience and observe OSNs' capability in assisting current steganographic techniques such as Least Significant Bit substitution (LSB), Discrete Cosine Transform (DCT) coefficients

and EOF appending technique in images with specific OSN features. The result of these tests is to determine which image steganographic tools or techniques a specific OSN's feature supports particularly in the Facebook and Google+ platforms. From these observations, the researcher will have an idea of the potential techniques that can be used for secret message embedding on OSNs.

Phase 2 is to develop two case scenarios that exploit the techniques found in Phase 1 in Facebook and Google+. The purpose of Phase 2 is to generate evidence for data collection as well as to establish control data as a baseline for analysis.

In Phase 3, data acquisition and extraction will be performed on the established case scenarios using the computer forensic guidance model suggested by Nouredin et al. (2011).

In Phase 4, the data extracted will be reconstructed and a comparative analysis will be conducted to compare the evidence identified with the control data.

All of the investigation steps will be documented in journal form, and lastly in Phase 5, an effective method of initiating an investigation which includes steganographic evaluation drawn from the experiment will be recommended.

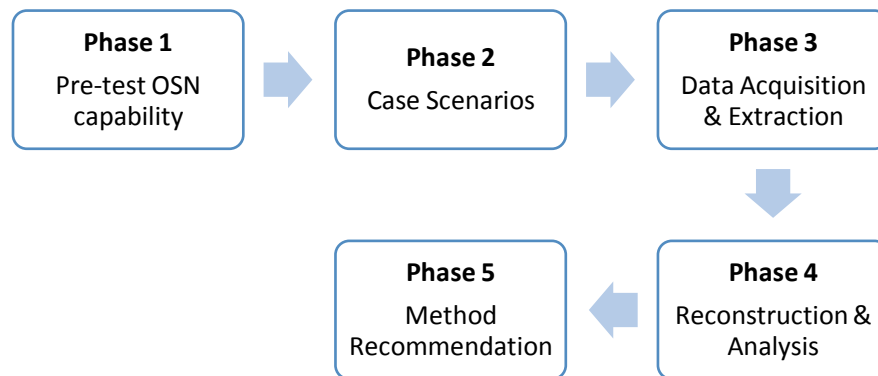


Figure 3.5: Research Phases

3.2.5 Data Map

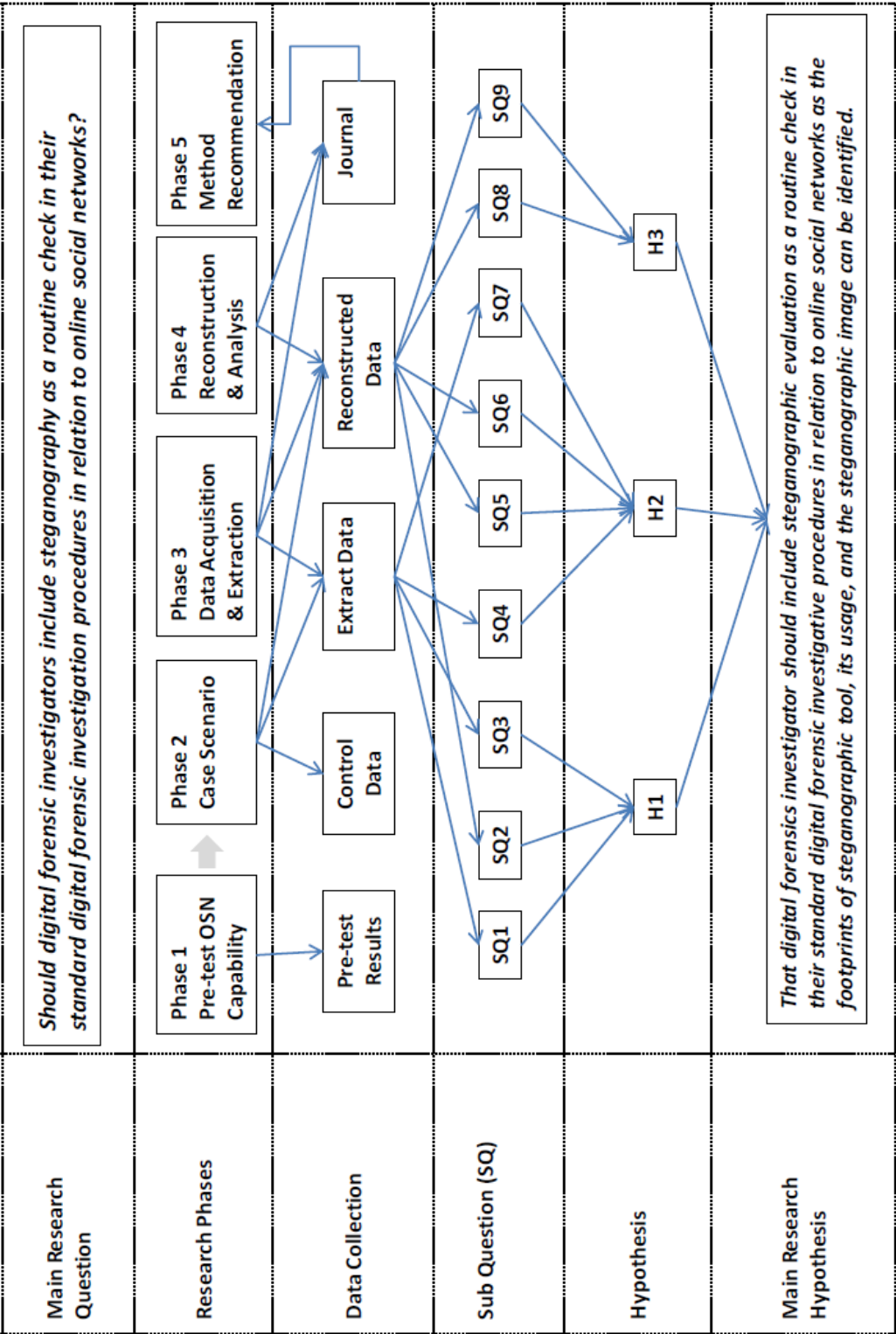


Figure 3.6: Proposed Research Data Map

3.3 DATA REQUIREMENTS

There are several sources of data that are required for the proposed research including pre-test data, control data, extracted data, reconstructed data, and the documented journal on the investigation. Pre-test results are needed in order to understand the capability of steganographic tools as well as how OSNs assist or inhibit image steganography. Control data is the sample evidence that is generated based on the fictitious scenarios, which are to portray as closely as possible a real world event. The control data is to be recorded into a table as the known or expected artefacts and will later be used as a comparative baseline for the artefacts extracted and reconstructed from the case scenario through a digital forensic process.

The second and third requirements for data are the extracted and reconstructed data. Before the extraction and reconstruction, the data is to be acquired from the target machine where the scenario is performed. Once all the required data are gathered, a comparative analysis with the control data will be conducted with the aim of answering the sub-questions and ultimately the main research question. The step by step investigation processes conducted on the experimental case scenarios will also be recorded in journal form to ensure that the steps are repeatable.

3.3.1 Investigation Case Scenarios

There are two case scenarios for simulation generated on two different OSN platforms. Experimental Case Scenario 1 is performed on Facebook where the incriminating activities are associated with terrorism while in Case Scenario 2 is performed on Google+ and is associated with corporate intellectual property theft. Two fictitious criminal characters, John Doe and Christian Riley were used to portray the sender and receiver for the covert communication in both the experimental case scenarios.

3.3.1.1 Terrorism - Case Scenario 1

Christian Riley was forced to become a terrorist or else his family would die, he needed to contact a certain person overseas to discuss about the next plan of attack, but he had no way of doing so because everyone was being watched and he was

being monitored on suspicion of criminal activity. Every communication channel was monitored and so were social media such as Myspace, Twitter, Facebook and Google+. Because there are too many people to check individually on OSN, it presented a possibility for communication. A platform where a privacy configuration is possible, where only friends can see and talk to each other, would also help.

John Doe is part of an organization that wants to terrorize people, he only needed to get the text files which had the next plan of attack; the only problem is he didn't know when the person was going to contact him. He waited patiently to hear from his boss about Christian Riley who is going to contact him. They were instructed to communicate and exchange information over social networking sites. They used Facebook for communication and for sending secret information. They used an image steganography method in which they sent each other normal images but within which were hidden text files that had a hidden message for each other. Both users' Facebook pages are accessible by invitation only and cannot be viewed by anyone else.

They added each other on Facebook and from there they blocked off anyone seeing them as friends or their wall posts and chats using the private group option in Facebook. Christian Riley initiated the conversation with John Doe. John Doe is told by the Christian Riley that he has shared images on his Facebook page and that he has added John Doe to the group so that only John Doe can download the images. A law enforcement team has picked up on the call as which they think is a possible threat and have hired a forensic examiner to look for evidence from their computer. The law enforcement officer has already seized the suspect's hard drive. A forensic investigator is given the tasks of extracting/analysing any potential evidence from OSNs on the suspect's HDD.

3.3.1.2 Intellectual Property - Case Scenario 2

Starworld is a hospitality company that owns more than 50 cafes and convenience stores in Auckland. John Doe, one of the marketing team members was very unhappy with the recent decision to promote Steven as Sales Manager instead of him. John Doe thinks that he deserved it more than Steven. So, to show his unhappiness at the company, he started sending the company's weekly unreleased promotional information and business's plans to a competitor, XO Mart.

Starworld company's IT policy blocks the USB port from saving files externally, so to send the confidential information to the competitor covertly, John Doe decided to use Google+ to communicate with the competitor and use image steganography to transport the confidential information rather than using email as sending photos via email would get the network administrator's attention if it were too frequent, whereas sharing photos in OSN is a more inconspicuous activity. Moreover, Starworld permits staff to use OSNs.

Therefore with the help from John Doe, XO Mart, located two blocks away, knows Starworld's insider plans and has taken on their competitor easily and this has impacted Starworld's businesses. Since the pattern was so persistent, the management team decided to undertake an internal investigation of the sales and marketing department as promotional items and price were planned and organized by the team. From an interview, Richard, the Sales and Marketing Director, told the investigation team that, John Doe had acted differently since Steven had been promoted Sales Manager last month and other colleagues also said that John Doe was telling other team members that he deserved better. One of them even saw John Doe was having coffee with the XO Mart Managing Director three days ago and the network administrator found that John Doe had been spending lots of his work time on Google+ lately.

From the interview, John Doe seems to be a suspect, thus, the company decided to seize John Doe's work computer and the hard drive was brought by the IT team to the forensic lab to look for evidence of John distributing confidential company information to XO Mart. Information collected from the interview was passed on to the forensic team, and the forensic team decided to look for any traces they can gather from Google+ as this was the most predominant activity that performed lately, and there were no suspicious emails reported by the network administrator.

3.3.2 Data Collection

The first data to be collected are the pre-test results from six different steganographic techniques or tools (JP Hide and Seek, SilentEye, EOF injection, StegHide, S-Tools, Invisible Secrets 4) tested on Facebook and Google+. JP Hide and Seek, SilentEye and EOF injection is to generate JPEG format steganographic image. StegHide is to generate BMP format steganographic image. S-Tools is to

generate GIF format steganographic image and Invisible Secrets 4 is to generate PNG format steganographic image. The steganographic images generated by the above mentioned steganographic techniques (6 images) will be uploaded using three different features on Facebook – photo upload, file sharing, and message attachment and using one feature (photo upload) on Google+. These uploaded images are then to be downloaded from Facebook and Google+ to see whether the embedded secret messages can be successfully extracted. This pre-test data will be able to ascertain which steganographic techniques and image formats can or cannot be used and which OSN features can assist or inhibit image steganography.

In order to collect the extracted data and reconstructed data, case scenarios activities depicted in Sections 3.3.1.1 and 3.3.1.2 will be simulated on the experimental machine in the lab environment. During this process, the simulated activities on the experimental machine will be recorded, these data are also known as control data. The variables that will be collected during this collection process are; the name of the steganography tool, activities performed, date and time of such activity, which cover image was used, what was hidden, and the MD5 values of the images uploaded onto the OSN. All the information will be recorded in table form.

Extracted data in Phase 3 of the experiment will be collected using forensically sound methods where a write blocker will be used to acquire the evidence from the set up target's machine. The purpose of using write blocker is to ensure that the data is imaged from the target machine without changing it and thereby ensuring and preserving data integrity. This process can be verified by matching the computed MD5 values after the acquisition process. The forensic tools that will be used in this process are the FTK Imaging tool and Encase Guidance Software. Once the acquisition process is complete, extracted data can be collected. Depending on the case, the forensic examiner will need to determine and identify what kind of information needs to be extracted and possibly make all information visible (Noureldin et al., 2011). According to Noureldin et al., “it is necessary to extract data that have been deleted, hidden, camouflaged, or that are otherwise unavailable for viewing using the native operating system and resident file system” (2011, p.566).

Reconstructed data is collected during the forensic analysis process, which is Phase 4 of the experiment. This is where the pieces of evidence are collected

and where missing pieces are found in order to create a picture of the criminal events (Noureldin et al., 2011). Reconstructed data can be used to determine what happened, when it happened, how it happened, where the evidence was found, why it happened and possibly who did it.

From collecting extracted data to reconstructing data, each step of the investigation and which tools were used will be reported in journal format. The information documented in the journal is vital, as it records all the procedures undertaken in the investigation. This is to ensure that the procedures are repeatable and are able to reproduce similar results and to recommend an effective investigation method for similar environments.

3.3.3 Data Processing

As mentioned earlier in Section 3.3.2, there are all together five types of data that are needed to be collected: pre-test data, control data, extracted data, and reconstructed data. All the collected data will be processed in a tabular form by using an Excel spreadsheet. This is to ensure that the collected data can be evaluated in an effective and concise way.

Control data is processed in a lab environment using two computers freshly installed with the Windows 7 (Professional Edition) operating system the hard drives having been wiped using Darik's Boot and Nuke (DBAN) wiping utility tool based on The American Department of Defence 5220.22.m short wipe standard. This wiping method is composed of passes 1, 2, 7 from standard wipe. Same process is undertaken for the pre-test machine as well as for each case scenario so that the data is cleaned of previous data. One computer will be the simulated sender machine and the other one will act as the receiver machine. Internet Explorer (IE) was chosen as IE is the pre-installed browser for the Windows system. All images downloading from the OSN will be saved under the default file name at the time of downloading, which means the user will download and save the file without changing the file name. Each activity, all evidence created, and all tools that are used in the scenario will be recorded and marked as known evidential artefacts. Subsequently these control data will be used for comparative analysis with the reconstructed evidence from the digital forensic investigation process.

Digital evidence is fragile; “it can be altered, damaged, or destroyed easily by improper handling or examination” (NIJ, 2004, p.11). To preserve and ensure the integrity of the digital evidence, all extracted and reconstructed data are processed with MD5 hash values before and after the analysis. When both MD5 values match, it is assumed that nothing has been altered during the analysis process, and thus confirms the reliability of the extracted and reconstructed evidence. The tools that would be used in this processing include a write blocker, Encase software, FTK Imager, CacheBack, StegAlyserAS, StegAlyserSS and any other tools necessary for extracting and reconstructing the evidence.

The journal documenting events during the investigation is an important set of data that can be used to recommend the best practice for forensic investigation procedures for steganography involved in online social networks. The documented steps of the investigation in the journal will be transferred into a simple and comprehensive flow chart diagram for easy interpretation.

3.3.4 Data Analysis

The data analysis of this proposed research is divided into three parts. First is the analysis of the pre-test results conducted in Phase 1. This is to analyze the OSN’s capability and features that support image steganographic activities on various common image formats such as BMP, JPEG, PNG, and GIF. The second part of data analysis is forensic analysis on the extracted and reconstructed data that have been collected and processed, as mentioned in Sections 3.3.2 and 3.3.3. Thirdly, a comparative analysis will be performed on the findings from the forensics investigation and the control data.

The pre-test results analysis is based on the two platforms, Facebook and Google+ on which the test is conducted. From the test results collected, the two platforms are compared in terms of the features that support image uploading, the formats that accept image steganography, and whether the hidden message can be successfully extracted from the downloaded images. Successful hidden message extraction from the downloaded images is important because as was seen in the literature review in Chapter 2, Section 2.4, OSNs pre-process the uploaded images before publishing them on the user’s OSN page and this action can possibly destroy the hidden message embedded in an image. Therefore, from analysis of

the pre-test results, image steganographic techniques and formats that are supported by each OSN is tested and verified.

The second part of the analysis is called forensic analysis. Forensic analysis will be performed on the extracted and reconstructed data that have been collected in Phases 3 and 4 of the experiment. Forensic analysis is to analyze and understand the extracted and reconstructed data into useful information that can be used as admissible evidence in a court of law. Forensic analysis consists of “timeframe analysis, data hiding analysis, application and file analysis, ownership and possession analysis, log files analysis, analysis of email messages and network analysis” depending on the cases (Noureldin et al., 2011, p.567). For the nature of this experiment, it is expected that forensic analysis such as data hiding analysis, file analysis will be involved. However, social network analysis and web browser analysis will be as well added into the analysis process. The results from these selected analyses will be able to map and interpret the pieces of evidence that are collected and reconstructed and therefore answer the question of what happened, when it happened, how it happened, where the evidence was found, why it happened and possibly who did it.

The third part of the analysis is a comparative analysis between the control data and the reconstructed data. The objective of this analysis is to determine whether the use of steganography tools and the steganographic images uploaded into the OSN or downloaded from OSN can be identified and whether the hidden messages can be extracted. If the results show a positive outcome that steganography tools and steganographic images can be identified, then according to the NIJ report, *Forensic examination of digital evidence: A guide for law enforcement*, such data “may indicate knowledge, ownership, or intent” (NIJ, 2004, p.17). Additionally, the result will ultimately answer the asserted main research hypothesis: “*That digital forensics investigator should include steganographic evaluation as a routine check in their standard digital forensic investigative procedures in relation to online social networks as the footprints of steganographic tool, its usage, and the steganographic image can be identified.*”

3.3.5 Data Presentations

The test data collected in Phase 1 of the experiment will be presented in a tabular form listing the features of the OSN, tools and techniques used, the file name and

MD5 value for each cover image, generated steganographic image, and the downloaded image, and finally the success of hidden message extraction will be indicated with a yes or no. Remarks will also be recorded if there is any additional relevant information from the observation.

The control data will be presented in a tabular form to indicate steganography tool used, steganographic images that used for uploaded or download on the online social networks, and any other associated social network activities such as chat, message posting will be recorded.

Extracted and reconstructed data will be presented mostly in table form generated by the digital forensic tools. As mentioned earlier, control data is also called the known or expected artefacts; therefore during the process of digital forensic analysis, if the expected artefacts are identified by the forensic tool, then the relevant data will be exported into a table in Excel for data reconstruction. These data will be collected until the end of the forensic process. If expected artefacts are found, then they will be recorded into the comparative analysis table as found, or partly found, and how they were found. Steps taken along the investigative process will also be recorded as a journal to complete documentation. Finally, a recommendation for an effective guideline for evaluating steganographic investigation will be established from the documentation and presented in an easy-to-understand flow chart diagram.

3.4 LIMITATIONS

There is no doubt that there will be some limitations encountered in the proposed research methodology, yet it is important to be able to recognize these limitations so that the findings of the proposed research can be justified without bias. Therefore, the objective of this section is to discuss the limitations of the proposed research methodology and also to identify any aspects that could be transferable to similar research areas.

The steganographic techniques used in the proposed experiment are limited. The experiment tests on image steganography while there are many more steganographic techniques that have not been tested and covered in the proposed research such as video and text steganography that could possibly be used on an OSN. Additionally, the tools that applied in this research are limited to six

steganographic techniques, however, there are more than three hundred steganographic applications available not to mention high tech criminals who are capable of writing their own steganographic programs which are, of course, unpublished. Hence, the findings of the experiment can only be implied to other techniques that are similar to the six techniques used in the proposed experiment. These six techniques were chosen for the proposed experiment because they are easy to access from a Google search, are easily downloaded from the Internet, have easy to use graphical interfaces, and are free to download, which makes them attractive to the general public including those with nefarious intent.

Secondly, the operating system set up for the experimental case scenarios use the Windows 7 environment with pre-installed Internet Explorer. Therefore, the forensic investigation methods used in this experiment may be different with other operating systems like Mac or Linux. These platforms may have different file systems or structures as compared to the Windows 7 platform. Moreover, the experiment is conducted on Internet Explorer, so the findings are limited to Internet Explorer, whereas in the real world, a criminal may use more than one web browser on a system.

Similarly, the two most popular OSN platforms, Facebook and Google+ are used to test and verify the OSN environment and the experimental case scenarios are performed on these two platforms. Although these two environments are the most popular ones, they may not generalize to all OSN providers as each OSN website has its own unique architecture for data representation. Therefore the investigation methods used in the proposed research can be transferable only to platforms that similar to Facebook and Google+. However, the approach to identifying and extracting in the experiment can be adopted for other OSN platforms.

Furthermore, the investigation techniques used in the proposed research are limited to a shut down system during seizure, so live forensics and network forensics are not included. Lastly, each forensic tool has its own capabilities and limitation. The experiment findings are based on the evaluation given by the selected tools in the project, which are Encase software, FTK Imager, Internet Evidence Finder, StegDetect, StegAlyzerAS and StegAlyzerSS.

3.5 CONCLUSION

Chapter 3 has given an overview of the proposed research design, which includes the research methodology, research question and sub-questions that need to be answered, research hypotheses developed for testing, research phases, data required for the research as well as the limitations encountered in the research. Similar works from previous researchers have also been studied in order to find the most appropriate methodology that can be adopted for the proposed research. The reports from previous researchers have guided the establishment of the research methodology and design.

The review of the problems and issues that were presented in Section 2.6 as well as the information that was learned from the literature review in Chapter 2 have assisted in selecting the research problem and formulating the research question. Subsequently, based on the refined research question, sub-questions and hypotheses that need to be tested were formed. Research phases (Figure 3.5) have also been developed based on the empirical approach so that the experiment can be observed and systematically processed. The research data map was also given to show mappings between the research phases and the associated research sub-questions and hypotheses.

Additionally, experimental case scenarios and data requirements for the proposed research were also clearly described and discussed in this chapter. Lastly, the research limitations were considered and discussed so that the research findings can be correctly evaluated. Chapter 3 has illustrated the selected research problem area as well as the research methodologies that will be implemented in the proposed research in order to accomplish the research objective. Chapter 4 is now to report the findings of the experiments that were defined in this chapter.

Chapter 4

Research Findings and Analysis

4.0 INTRODUCTION

Chapter 3 established a research methodology for investigating steganographic techniques for online social networks (OSNs) and procedures for digital forensic investigation in this context. Relevant studies from previous research were selected for review and guided the proposed research methodology. The research question, sub-questions as well as the research hypotheses were then derived for the selected problem and issues that were identified in the literature review in Chapter 2. The data requirements for the experimentation were presented and the limitations of the proposed research discussed.

Chapter 4 however is to report the findings of the research phases defined in Chapter 3. Any variation between the outlined methodologies and the actual experimentation will be discussed in Section 4.1. The findings from data collection, data processing and data analysis will be presented in Sections 4.2, 4.3 and 4.4. Section 4.2 is to report the findings of various steganographic techniques that can be exploited on Facebook and Google+ whereas Sections 4.3 and 4.4 are to report the findings of digital forensic investigation on the experimental case scenarios defined in Section 3.3.1.

4.1 VARIATIONS ENCOUNTERED

It is inevitable that some unforeseen circumstances may be encountered during the actual experiment. It is important to report the variations that were encountered in the experiment such as the changes in the scenario environment, data collection, or data analysis as this may affect the outcome of the research findings

4.1.1 Case Scenario

There were some changes to Case Scenario 1 (Section 3.3.1.1) during the simulation with the experimental system. The simulation had to be performed twice as the first simulation process failed to gather Facebook chat data, which

was to be leading evidence in the Scenario 1. The first simulation process failed because Facebook chat no longer left its artefacts in the browser's cache file. Facebook chat artefacts are now left mainly in the pagefile and hibernation files. Although Facebook chat can be found in pagefiles and the hibernation files, for the first simulation process, Facebook chat artefacts were remained unrecovered as the simulation process was done too quickly and the data in the RAM was yet to be swapped over to pagefiles and hibernation during the process. The other way to find Facebook chat is from a memory dump, however, in the first simulation, memory dump was not performed, so none of the Facebook chat could be recovered. Consequently, the simulation of Scenario 1 had to be reprocessed for data collection. Again, both experimental hard drives were wiped and reinstalled with the Windows 7 operating system. Data uploaded into the experimental Facebook account in the first simulation were deleted to ensure that the current data was not mixed with the previous data.

In the second simulation process for Case Scenario 1, the experimental system was run overnight and the activities performed on Facebook were spread over two days. The system also went through some hibernation and other activities not related to the case scenario. Other activities were also performed like browsing other web sites, opening up other applications and so on. The variations occurred to ensure that the required experimental data could be collected and the scenario simulated as closely as possible a real world environment.

Case Scenario 1 and Case Scenario 2 in Sections 3.3.1.1 and 3.3.1.2 did not indicate clearly how the hard drive was acquired by the first responder. It is necessary to document precisely as to whether the system was dead or alive when the first responder seized the hard drive. Therefore, the following additional information is added to Case Scenario 1 – The suspect's system was running when the law enforcement officer seized the computer, memory dump was captured by law enforcement and the hard drive was seized by pulling the power cord from the live system. The suspect's laptop in Case Scenario 2 on the other hand was seized in a shut down condition.

4.1.2 Data Collection

As indicated in Section 3.3.2, there are actually three vital parts to data collection: pre-test data, evidence data (extracted data and reconstructed data), and control

data. In the actual experiment, there was some variance in pre-test data collection. Additional data was collected for pre-test data due to the inconsistent performance of a steganography tool – SilentEye. It was outlined in section 3.3.1 that there will be six images uploaded with on Facebook and Google+ feature. However, expected secret message extraction on the downloaded steganographic image from Facebook failed due to the luminance interval configuration of SilentEye. Therefore, an additional steganographic image with a different luminance level was created in order to show the possibility of using SilentEye to exploit the Facebook photo upload feature. Consequently, there will be seven images instead of six images for this particular feature.

Images that were planned to be used for Case Scenario 1 data collection were also having some problems with the StegDetect tool. These images were previously captured on a Canon IXUS digital camera and resized to 640 x 480 pixels, but when using StegDetect to analyze these images, some of them showed error messages “Quantization table 0x00 was not defined” or “Quantization table 0x01 was not defined” even though the images can be viewed normally and can be embedded with a secret message. Thus, the images for Case Scenario 1 were specifically chosen to ensure that there were no errors in quantization table for the experiment and to make sure that these images were detected by StegDetect as clean images (negative) before the simulation for data collection.

RAM acquisition for Case Scenario 1 was also added to data collection during the actual experiment. This data were collected for backup purposes so that viable evidence which may have been left in the RAM, especially those social networking activities associated with live chat, could be recovered in case these data had not yet been swapped to the pagefile in the system.

4.1.3 Data Processing

A hardware write-blocker was proposed in the methodology in order to preserve and ensure the integrity of the data during the acquisition process. The SATA hard drive was connected to the ‘SATA to USB’ connector and, using a USB cable the SATA hard drive is able to connect to the TABLEAU T8 – Forensic USB Bridge write blocker. This was tested before the actual experiment and it worked correctly. However, an unexpected situation occurred after the disk wiping. TABLEAU T8 no longer recognized the hard drive. In order to resolve this issue,

the hard drive was plugged in directly into the USB drive of the investigator's PC and the hard drive was detectable. Therefore, a software write blocker called SAFE Block Win 7 by ForensicSoft was used in the actual experiment instead of the TABLEAU T8 hardware write blocker.

There was an additional process for data extraction. In the proposed methodology, RAM memory dump data processing was not included. However, in the actual experiment, a RAM memory dump was collected. The memory dump data was processed with FTK Imager software and EnCase software so that necessary data can be viewed and extracted. There were also some changes to the software that was used for forensic analysis. FTK Imager was used for the bit to bit acquisition (imaging) as well as RAM memory acquisition. EnCase version 7.0, Internet Evidence Finder 5.0 (IEF v5), WinPrefetchView, StegAlyzerAS, StegAlyzerSS, and StegDetect were used for analyzing the case scenarios. Internet Evidence Finder 5.0 was used in the experiment instead of CacheBack due to the availability of the tool and StegDetect was added for additional data processing.

4.1.4 Data Analysis and Presentation

There were no major changes to data analysis and data presentation as per Sections 3.3.4 and 3.3.5. However, some screen shots of data analysis were added for completeness.

4.2 SOCIAL MEDIA PRE-TEST

The purpose of the social media pre-test was to identify the possible ways of performing image steganography on the two popular social media platforms: Facebook and Google+. Both platforms have their own unique architecture and user interface layout. Figures 4.1 and 4.2 are the home page layouts of Facebook and Google+. There are three ways a user can share their image files on Facebook. The images can be shared through regular photo upload, group files sharing, or as an attachment in a message, whereas Google+ has only one way to share images, which is via regular photo upload. Therefore, in this pretest, six steganographic techniques, as mentioned in Section 3.3.2: JP Hide and Seek, SilentEye, EOF injection, StegHide, S-Tools, and Invisible Secrets 4 have been tested with the available methods of photo sharing features in Facebook and Google+.



Figure 4.1: Facebook Home Page Layout

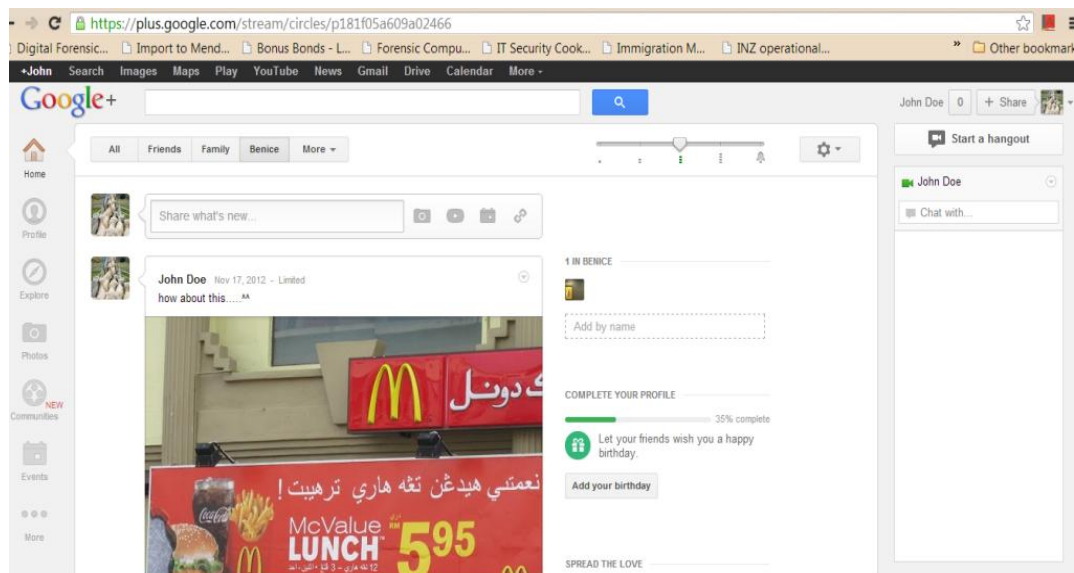


Figure 4.2: Google+ Home Page Layout

4.2.1 Environment Set Up

A laptop equipped with Wifi connection, Intel® Core™ 2 Duo CPU, 2.20GHz, 2GB RAM and 500GB hard drive was used for this testing. Both Facebook and Google+ platforms were used for this test. The process of steganography for social media was set up as in Figure 4.3. The photos used in the test was taken on a Canon IXUS 110 IS digital camera, edited into size 480 X 640 pixels and labeled a unique name from FB_P1 to FB_P18 for Facebook, and from G_P1 to G_P6 for Google+. This unique label includes the information about which social

media platform, which sharing feature, and which steganographic technique was used. For example FB_P1 means the cover image is for JP Hide and Seek to generate steganographic image for Facebook photo upload feature. Once the steganographic process is completed the steganographic image will be saved into a file name starts with an ‘S’ followed by the same unique label’s name. For example, cover-object labeled with FB_P1 will be saved as SFB_P1. The full list of the identifier can be found in Appendix 8 and Appendix 9. The lab environment for the process of steganography in social media is illustrated in Figure 4.3.

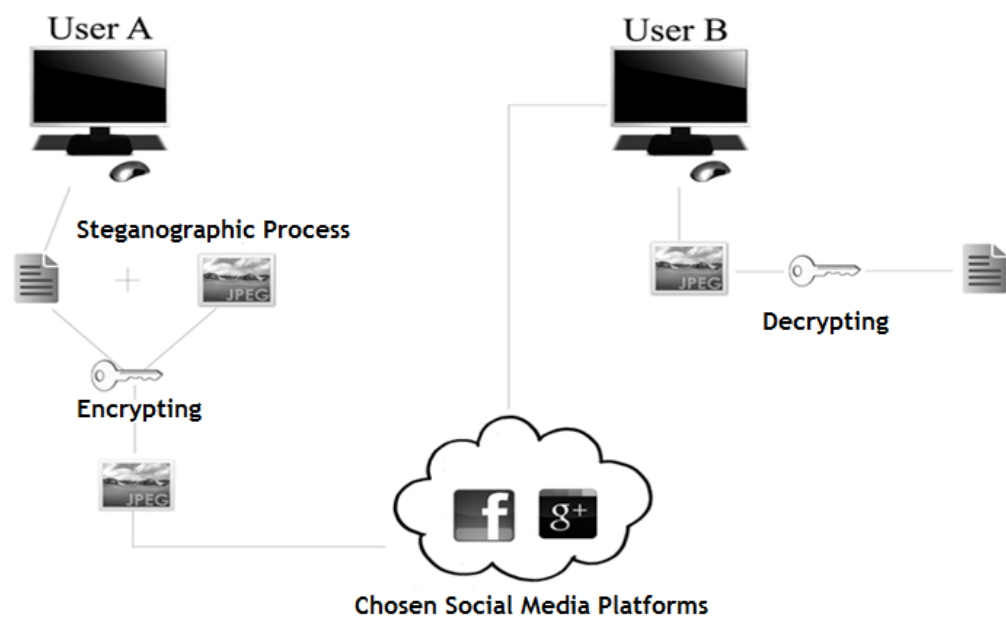


Figure 4.3: Lab environment steganographic process

4.2.2 Findings

Referring to Table 4.1, of the six steganographic techniques tested none of the tools was robust against Facebook photo publishing preprocesses in the photo upload feature except SilentEye. However, the success of secret message extraction for SilentEye is inconsistent. Based on the two images tested for the possibility of secret message extraction being 50%, the success of secret message extraction after download from Facebook photo upload feature depends on the luminance interval configured during the embedding process. The standard luminance interval 5, which was configured for all the other tests in the pre-test, failed for secret message extraction after the download. But, when the luminance

interval was increased to 10, the secret message was successfully extracted after the download. It has already been pointed out that the Facebook photo upload feature will preprocess and change other image formats to JPEG before publishing it on Facebook. Therefore, this preprocess inhibits other possible image steganography on Facebook photo upload feature for techniques that use BMP, PNG, and GIF as their cover images.

It was also found that, even though the secret message embedded using SilentEye can be successfully extracted and the content of the secret message before upload and after download are the same, the MD5 value of the downloaded steganographic image was different from the steganographic image used for photo upload. This is due to Facebook's photo publishing preprocesses changing the integrity of the uploaded images. It is also necessary to note that Facebook allocates its own file name to the uploaded photo. For example, the uploaded photo named as SFB_P2.jpg by the user will be renamed by Facebook as 149889_168496316622410_84868167_n.jpg when published on Facebook. Therefore, when the photo download is performed, by default, 149889_168496316622410_84868167_n.jpg will be shown in the file saving dialog box for the user to save (Figure 4.4). However, users can still download and save the file with their own preferred file name.

Table 4.1: Facebook Photo Upload Results

Facebook Photo Upload			
Steganographic Tools	Steganographic Image Uploaded	Secret Message Extracted	Success Rate
JP Hide and Seek	1	0	0.00%
Silent Eye*	2	1	50.00%
EOF	1	0	0.00%
StegHide	1	0	0.00%
S-Tools	1	0	0.00%
Invisible Secret 4	1	0	0.00%

* The successful secret message extraction is from steganographic images that were configured to luminance interval 10 and 30% photo quality on the chosen image

Image steganography is highly feasible in Google+. 100% of the tested pictures that were embedded with a secret message using various steganography tools can successfully extracted after download (Table 4.2). It is evident that Google+ accepted various steganographic techniques and image formats, JPEG,

BMP, GIF and PNG. It also confirmed that Google+ does not make any changes to the uploaded steganographic image as the MD5 values before upload and after download were the same. The details of the data collected can be found in Appendix 11.

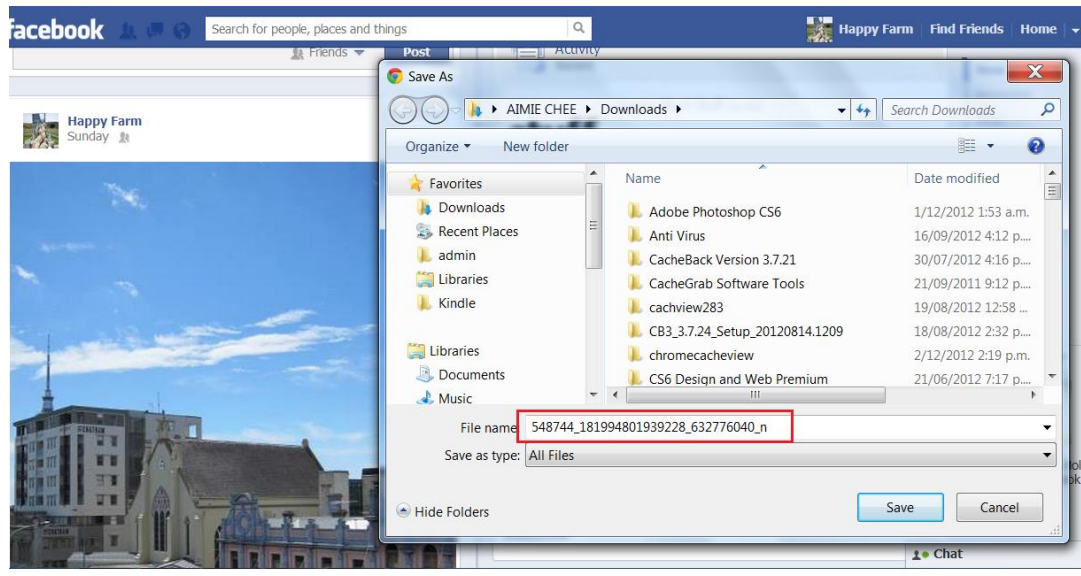


Figure 4.4: Facebook default picture file name when download

Table 4.2: Google+ Photo Upload Results

Google+ Photo Upload			
Steganographic Tools	Steganographic Image Uploaded	Secret Message Extracted	Success Rate
JP Hide and Seek	1	1	100.00%
Silent Eye*	1	1	100.00%
EOF	1	1	100.00%
StegHide	1	1	100.00%
S-Tools	1	1	100.00%
Invisible Secret 4	1	1	100.00%

* The luminance interval configuration is 5 and photo quality 30% on the chosen image

Another option for sharing images on Facebook is to use the file sharing feature that is available in Facebook group (Figure 4.5). All shared files in Facebook can be listed in the file section as shown in Figure 4.6. This feature is significantly preferable for steganographic purposes as compared with the photo upload feature. The experimental results showed that the six different steganographic techniques when using the file sharing feature on Facebook had a 100% success rate in secret message extraction after download (Table 4.3). Furthermore, steganographic

techniques using BMP, GIF and PNG formats are also accepted in the file sharing feature on Facebook. It was also found that, there were no changes on the shared steganographic images before upload and after download as both MD5 values are the same. Details of the collected data can be found in Appendix 10.

Table 4.3: Facebook File Sharing Results

Facebook File Sharing			
Steganographic Tools	Steganographic Image Uploaded	Secret Message Extracted	Success Rate
JP Hide and Seek	1	1	100.00%
Silent Eye*	1	1	100.00%
EOF	1	1	100.00%
StegHide	1	1	100.00%
S-Tools	1	1	100.00%
Invisible Secret 4	1	1	100.00%

* The configuration of luminance interval is at 5 and photo quality at 30% on the chosen image



Figure 4.5: Facebook group file sharing feature

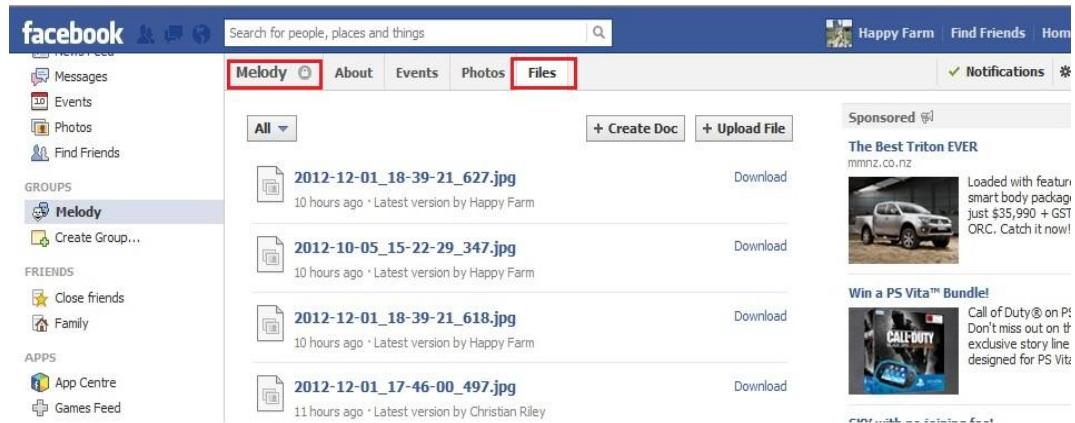


Figure 4.6: All the files that the user shared with the Melody group

The third feature in Facebook that enables the user to share steganographic images with an intended recipient is the attachment feature in Facebook messages (Figure 4.7). There are two types of attachment that can be shared in messages, Add Files and Add Photo. In order to avoid Facebook photo publishing preprocesses, the user can choose the Add Files attachment instead of Add Photo attachment. By choosing the Add Files attachment, steganographic images can bypass the Facebook photo publishing preprocesses that can destroy the secret messages and successfully share secret message with the intended recipient. The pre-test has proved that all the steganographic techniques tested with this feature had a 100% success rate for secret message extraction (Table 4.4) and the MD5 values for both the uploaded and downloaded steganographic images are the same. Additionally, the file name of the attachment remained as it was without any changes to the file name after being downloaded.

Table 4.4: Facebook Message Attachment Results

Facebook Message Attachment			
Steganographic Tools	Steganographic Image Uploaded	Secret Message Extracted	Success Rate
JP Hide and Seek	1	1	100.00%
Silent Eye*	1	1	100.00%
EOF	1	1	100.00%
StegHide	1	1	100.00%
S-Tools	1	1	100.00%
Invisible Secret 4	1	1	100.00%

* The luminance interval configuration at 5 and photo quality at 30% on the chosen image

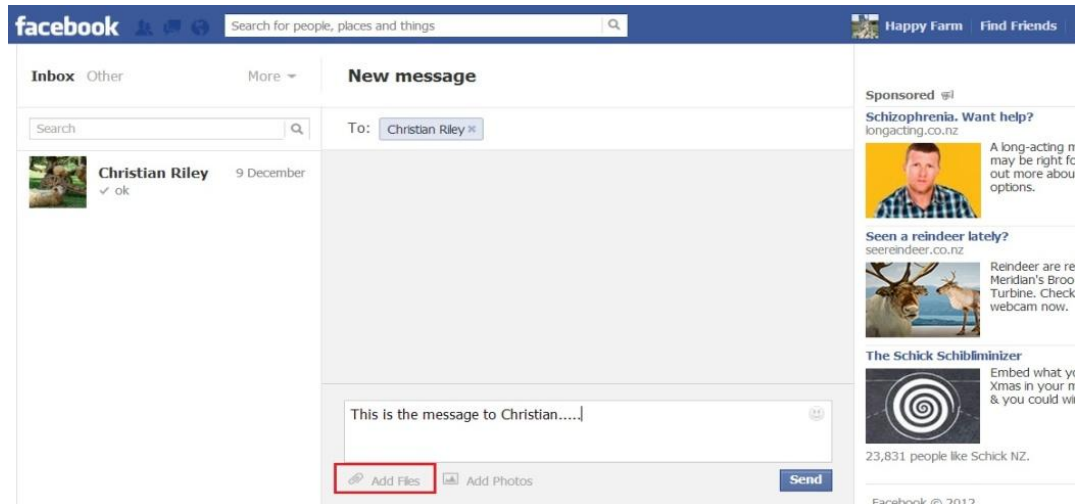


Figure 4.7: Steganographic images can be attached to the message by using the Add Files function

4.2.3 Social Media: Steganographic Techniques

This section reports the various steganographic techniques that are accepted and inhibited by Facebook and Google+.

4.2.3.1 Facebook

To summarize, the most common steganographic tools that can be found on the internet will not be able to generate steganographic images that are robust against Facebook photo publishing preprocesses. However, image steganography can still be performed on Facebook through the file sharing and message attachment that are available on Facebook. Table 4.5 summarizes the steganographic techniques that can and cannot be used with a specific feature in Facebook. The details of the data collected during the experiment can be found in Appendix 10.

4.2.3.2 Google+

Google+ supports various image formats with different steganographic techniques in its photo upload feature. The flexibility in Google+ photo upload feature is favorable for steganographic processes because sharing images with photo upload is a common sharing activity in social media. Furthermore, photo upload is the only way to share photos in Google+. Therefore, this activity is unlikely to generate attention as compared to steganographic photos that are shared with file sharing and message attachment in Facebook. SilentEye may not be a desirable

tool for steganography as it generates obvious embedding artefacts on the cover image, which will easily arouse an adversary's attention (Figure 4.8). Table 4.6 summarizes the steganographic techniques that are supported by Google+ and the details of the data collected during the experiment can be found in Appendix 11.

Table 4.5: Steganographic techniques supported or inhibited on Facebook

Facebook				
Features	Tools used	Format Used	Successful Extraction Secret Message	
			Yes	No
Photo Upload	JP Hide and Seek	JPEG		√
	Silent Eye*	JPEG		√
	EOF	JPEG		√
	StegHide	BMP		√
	S-Tools	GIF		√
	Invisible Secrets 4	PNG		√
File Sharing	JP Hide and Seek	JPEG	√	
	Silent Eye	JPEG	√	
	EOF	JPEG	√	
	StegHide	BMP	√	
	S-Tools	GIF	√	
	Invisible Secrets 4	PNG	√	
Message Attachment	JP Hide and Seek	JPEG	√	
	Silent Eye	JPEG	√	
	EOF	JPEG	√	
	StegHide	BMP	√	
	S-Tools	GIF	√	
	Invisible Secrets 4	PNG	√	

Note: * Luminance Interval was set at 5 and JPG quality was configured to 30%

Table 4.6: Steganographic techniques supported in Google+

Google+				
Features	Tools used	Format Used	Successful Secret Message Extraction	
			Yes	No
Photo Upload	JP Hide and Seek	JPEG	√	
	Silent Eye*	JPEG	√	
	EOF	JPEG	√	
	StegHide	BMP	√	
	S-Tools	GIF	√	
	Invisible Secrets 4	PNG	√	
File Sharing	JP Hide and Seek	JPEG	Not Applicable	
	Silent Eye	JPEG		
	EOF	JPEG		
	StegHide	BMP		
	S-Tools	GIF		
	Invisible Secrets 4	PNG		
Message Attachment	JP Hide and Seek	JPEG		
	Silent Eye	JPEG		
	EOF	JPEG		
	StegHide	BMP		
	S-Tools	GIF		
	Invisible Secrets 4	PNG		

Note: * Luminance Interval was set at 5 and JPG quality was configured to 30%

4.2.4 Conclusion

In conclusion, Facebook has more functionality than Google+ in terms of image sharing capability. But, when comparing the supported steganographic techniques, Google+ would be preferable because all steganographic techniques tested could be used on Google+ for common photo sharing whereas not all the techniques work on Facebook due to its photo publishing preprocesses. There is the possibility of using SilentEye which may resist photo publishing preprocesses, but a user will be unlikely to use it as it creates an obvious embedded artifact in its cover image while other steganography tools that have similar secret messages embedding capabilities, in JPEG format, such as JP Hide and Seek, EOF, S-Tools, StegHide and Invisible Secrets 4 do not create any perceivable artefacts.

Furthermore, as compared to the other tools tested, SilentEye is not stable and is inconvenient to use. Each time a steganography is to be performed, its

luminance interval and photo quality configuration has to be tested; otherwise secret message extraction by the recipient may fail. The experimental experience also found that each time the user uses a different cover image for embedding or a different secret message capacity, this can also affect secret message extraction.

Therefore, if image steganography is to be performed on Facebook, it will most likely use file sharing and message attachment rather than a photo upload. If the user wanted to share steganographic images with photo upload, SilentEye is a steganographic technique that could be used; but, the obvious artefacts could be identified easily. On the other hand, if image steganography is to be performed on Google+, photo sharing is the only way to share it. Obvious artefacts can offer be seen in an image generated by SilentEye; otherwise most of the steganographic images generated by the tools tested would be hard for the human eye to identify.

Figure 4.8: Steganographic image generated by JP Hide and Seek (left) and



steganographic image generated by SilentEye (right)

4.3 CASE SCENARIO 1 - TERRORISM

The first case scenario is about covert communication between two terrorists on Facebook using image steganography. The objective of the investigation is to extract and analyze any potential evidence on the suspects' HDD associated with image uploading or downloading on Facebook. It is expected that the

steganographic tool's artefacts can be detected, steganographic images shared on Facebook can be identified and the secret message can be extracted.

4.3.1 Environment Set Up

Scenario 1 was set up with two laptops with Wifi connection. One laptop was a Pentium® Dual-Core™ CPU, 2.30GHz with 4GB RAM and 120GB hard drive (Target Machine 1 – Christian Riley). The other laptop was an Intel® Core™ 2 Duo CPU, 2.20GHz with 2GB RAM and 120GB hard drive (Target Machine 2 – John Doe). Both hard drives were fully wiped with Darik's Boot and Nuke (DBAN) data wiping utility to ensure the hard disks did not contain any previous data. Windows 7 Professional was installed on both hard drives. The photos used in Case Scenario 1 were captured on a Motorola MB525 mobile phone camera.

As for the forensic investigation environment, data collection of the target's hard drive was performed with a software write blocker called SAFE Block Win 7 and FTK Imager 3.0. The data acquisition setup is depicted in Figure 4.9 and all the acquired evidence image files were verified with MD5 and SHA hash values and saved in Encase evidence file format (.E01) on an external 1TB hard drive.



Figure 4.9: Data Acquisition Process

4.3.2 Digital Forensics

The digital forensic process is a critical process. Any mishandling in the process may invalidate the collected evidence and it may not be admissible in a court of law. Therefore, the digital forensic process conducted in the proposed experimental case scenarios was adapted from Noureldin, Hashem, and Abdalla

(2011) as discussed in Chapter 3. The process steps include: 1) Evaluation and Assessment 2) Acquisition of Digital Evidence 3) Survey of Digital Scene 4) Digital Evidence Examination 5) Reconstruction of Extracted Data 6) Conclusion.

4.3.2.1 Evaluation and Assessment

- Both suspects' laptops were powered on when seized.
- Memory dump was acquired during the seizure and saved as memdump.mem on law enforcement portable hard drive
- Both laptops' battery were then pulled off and sent to the lab
- Suspects' hard drives need to be taken off the seized laptop
- Tools needed: SATA to USB connector, software write blocker SAFE Block Win7, FTK Imager 3.0, Encase 7.0, Internet Evidence Finder, WinPrefetchView, StegAlyzerAS, StegAlyzerSS, StegDetect

4.3.2.2 Acquisition of Digital Evidence

- Take off suspects' hard drives from seized laptops
- Both hard drives are Western Digital hard drives
- Model: WD1200BEVS
- Storage: 120GB
- Serial Number 1: WXCZ07003402
- Serial Number 2: WXEZ07L46465
- Memory dump file 1: memdump.mem
- Memory dump file 2: memdump.mem

Each of the suspects' hard drives were connected to the investigator machine with a SATA to USB connector and were acquired one by one. The investigator machine that installed with SAFE Block Win 7 software write blocker and FTK imager was used to image the suspect's hard drive bit by bit and saved into an external hard drive as CRiley_Test2.E01 and JDoe_Test2.E01. This .E01 is called the evidence file or image file, which is an exact duplicate copy of suspects' hard drives. The integrity of CRiley_Test2.E01 and JDoe_Test2.E01 files were verified with MD5 and SHA values (Appendix 3 & Appendix 4). After both hard drives were successfully acquired, each RAM memory dump file - memdump.mem acquired by the first responder from both live machines were also imaged and saved as Test2_liveMemory_cRiley.E01 and Test2_liveMemory_JDoe.E01

(Appendix 5 & Appendix 6) for further extraction and analysis and the original RAM memory dump file and the physical hard drives were kept in a safe place.

4.3.2.3 Survey of Digital Scene

In this process, the suspect's level of technological competency is evaluated. Both suspects' imaged hard drives were mounted in StegAlyzerAS and StegAlyzerSS to search for steganographic tool artefacts and steganographic images. The evaluation of each of the imaged hard drives found two applications containing unique steganographic file artefacts, 0 signature files, 18 appended image files, and one file having LSB embedding (Table 4.7).

Table 4.7: StegAlyzerAS and StegAlyzerSS Detection Summary (Scenario 1)

Forensic Tool	Steganographic Artefacts Detected	No. Applications Found	
		CRiley_Test2.E01 (HDD)	JDoe_Test2.E01 (HDD)
StegAlyzerAS	Unique File Artefacts	2	2
StegAlyzerSS - Signature Analysis	Signature Artefacts	0	0
StegAlyzerSS - Append Analysis	Appended Artefacts	18	18
StegAlyzerSS - LSB Analysis	LSB Artefacts	1	1

4.3.2.4 Digital Evidence Examination

The hard drive and memory dump evidence files were entered in Encase 7.0 for data extraction and evidence processing. In this process, each file in the evidence file was hashed with MD5 and SHA to ensure the integrity of the data while forensic extraction and analysis is performed. Internet activity found in the process was also automatically extracted. Internet Evidence Finder was also used to extract data related to Facebook internet activities. Table 4.8 is a summary of the data extracted from the targets' hard drives and RAM memory dump.

Table 4.8: Summary of Facebook Related Internet Activities

Forensic Tool	Domain	No. of URLs Visited			
		CRiley_Test2.E01 (HDD)	Test2_liveMemory_cRiley.E01 (RAM)	JDoe_Test2.E01 (HDD)	Test2_liveMemory_JDoe.E01 (RAM)
Encase v7	Facebook.com	20	10	11	13
	Facebook.com/	122	87	102	107
	attachment.fbsbx.com/	17	26	36	36
IEF v5	Facebook.com/	356	162	105	266
	attachment.fbsbx.com/	20	13	18	19
	Facebook chat	19	0	11	1

Since there was a possibility of steganographic involvement, the Windows prefetch data was previewed in EnCase to determine the execution of such applications. Windows prefetch files can usually be found in C:\windows\Prefetch\ for Windows XP and subsequent versions. A prefetch file (*.pf) is created by Windows each time a user executes an application and all files loaded by a particular application are recorded in this prefetch file. This file contains useful information such as the last launched timestamp, files that loaded during execution, how many times the application has executed and so on (Carvey, 2012). It was found that the prefetch file, JPHSWIN.EXE-A941F80B.pf was created on 09/12/12 03:14:53pm and was last written on 10/12/12 11:04:53am on Christian Riley's machine and JPHSWIN.EXE-896E3C85.pf was created on 09/12/12 04:00:08pm and was last written on 10/12/12 01:29:26am on John Doe's machine. Please note that JPHSWIN.EXE is the execution file for JP Hide and Seek application. These two files were then exported for further extraction to the WinPrefetchView utility program to extract the information contained in the prefetch file. It was found that JPHSWIN.EXE in Christian Riley's machine was executed eight times and the last run time was on 10/12/2012, 11:04:42am (Figure 4.10). JPHSWIN.EXE on John Doe's machine was executed 11 times, and the last run time was on 10/12/2012, 01:29:16am (Figure 4.11).

Filename	Created Time	Modified Time	Last Run Time	File Size	Process EXE	Run Counter	Process
JPHSWIN.EXE-A941F80B.pf	9/12/2012 3:14:53 p.m.	10/12/2012 11:04:53 a.m.	10/12/2012 11:04:42 a.m.	57,634	JPHSWIN.EXE	8	D:\USER

Filename	Full Path	Device Path
IERTUTIL.DLL	D:\WINDOWS\SYSTEM32\IERTUTIL.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\I
IMAGERES.DLL	D:\WINDOWS\SYSTEM32\IMAGERES.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\I
IMAGERES.DLL.MUI	D:\WINDOWS\SYSTEM32\EN-US\IMAGERES.DLL.MUI	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\E
IMM32.DLL	D:\WINDOWS\SYSTEM32\IMM32.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\I
JPHSWIN.EXE	D:\USERS\CHRISTIAN\DOCUMENTS\CHRISTIAN\JPHS_05\JPHS05\JPHSWIN.EXE	\DEVICE\HARDDISKVOLUME2\USERS\CHRISTIAN\DOC
KERNEL32.DLL	D:\WINDOWS\SYSTEM32\KERNEL32.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\K
KERNELBASE.DLL	D:\WINDOWS\SYSTEM32\KERNELBASE.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\K

Figure 4.10: Prefetch files extraction on Christian Riley's machine

Filename	Created Time	Modified Time	Last Run Time	File Size	Process EXE	Run Counter	Process Path
JPHSWIN.EXE-896E3C85.pf	9/12/2012 4:00:08 p.m.	10/12/2012 1:29:26 a.m.	10/12/2012 1:29:16 a.m.	58,276	JPHSWIN.EXE	11	D:\USERS\JOHN\DOWNL

Filename	Full Path	Device Path
IMM32.DLL	D:\WINDOWS\SYSTEM32\IMM32.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\IMM32.DLL
JPHSWIN.EXE	D:\USERS\JOHN\DOWNLOADS\JPHS_05\JPHS05\JPHSWIN.EXE	\DEVICE\HARDDISKVOLUME2\USERS\JOHN\DOWNLOADS\JPHS_05\JPH
KERNEL32.DLL	D:\WINDOWS\SYSTEM32\KERNEL32.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\KERNEL32.DLL
KERNELBASE.DLL	D:\WINDOWS\SYSTEM32\KERNELBASE.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\KERNELBASE.DLL
LINKINFO.DLL	D:\WINDOWS\SYSTEM32\LINKINFO.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\LINKINFO.DLL
LOCALE.NLS	D:\WINDOWS\SYSTEM32\LOCALE.NLS	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\LOCALE.NLS
LPK.DLL	D:\WINDOWS\SYSTEM32\LPK.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\LPK.DLL
MFC42.DLL	D:\WINDOWS\SYSTEM32\MFC42.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\MFC42.DLL
MFC42.DLL.MUI	D:\WINDOWS\SYSTEM32\EN-US\MFC42.DLL.MUI	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\EN-US\MFC42.DLL

Figure 4.11: Prefetch file extraction on John Doe's machine

Based on the data extracted by IEF, there was little Facebook chat recovered. Facebook often changed its data handling. Previously, as stated in Section 2.5.1, Facebook chat artefacts can be found in the cache file with a pattern *p_[number string]=[number][1].txt*. However, during this experiment, this Facebook chat artefact pattern mentioned could not be found in the browser cache file. Therefore, a keyword search in EnCase was performed in order to extract more data associated with Facebook Chat. Figure 4.12 shows part of the results of the keyword search found on Christian Riley's machine.

[illegible]

Figure 4.12: Keyword Search extracted from Christian Riley’s imaged hard drive

Since November 2010, Facebook chat is automatically saved in a user's Facebook message inbox (Constine, 2010). So, Facebook conversation can sometimes still be found in the cache file, but in a different pattern. A majority of the time, Facebook live chat can be found in pagefile.sys, hiberfil.sys or RAM. Additionally, the extracted data from IEF also showed that Christian Riley's Facebook user name was Christian Riley, his Facebook ID number was 100003867343997 and John Doe had a nick name, Happy Farm and his ID number was 100003861284061. Based on these ID numbers, activities performed by the suspects on Facebook could be easily identified. The URL, <http://www.Facebook.com/groups/172888169522216> was also found in both suspects' imaged hard drives. This is the Facebook group user URL and the number 172888169522216 at the end of the URL is the Facebook group ID number.

4.3.2.5 Reconstruction of Extracted Data

In this section, the data extracted will be reconstructed in order to provide a better picture of the possible crime and any missing information. The keyword search on Facebook chat extracted previously was reconstructed into logical order. Significantly, there was a conversation in regard to information hiding between Christian Riley and John Doe. Table 4.9 shows the reconstructed Facebook chat between Christian Riley, Facebook user ID 100003867343997 and John Doe, Facebook ID, 100003861284061 extracted mainly from pagefile.sys.

Table 4.9 Reconstructed Facebook Chat

UTC (Converted Local Time)	fbid Facebook ID	Chat Content
Sun, 9 Dec 2012 15:40:24 +13:00	100003861284061	hi Christian I have downloaded it What's nex?
Sun, 9 Dec 2012 15:42:38 +13:00	100003867343997	great! Now go to this website http://linux01_gedg.de/~alatham/stego.html
Sun, 9 Dec 2012 15:43:04 +13:00	100003867343997	download the window version
Sun, 9 Dec 2012 15:45:15 +13:00	100003867343997	you need this software to get what you wanted
Sun, 9 Dec 2012 15:45:19 +13:00	100003861284061	Ok
Sun, 9 Dec 2012 15:48:47 +13:00	100003861284061	ok got the software
Sun, 9 Dec 2012 15:49:32 +13:00	100003867343997	do you think you know how to use it?
Sun, 9 Dec 2012 15:50:22 +13:00	100003867343997	it's pretty simple
Sun, 9 Dec 2012 15:51:21 +13:00	100003861284061	yes I guess so, but I think I need something to
Sun, 9 Dec 2012 15:53:01 +13:00	100003867343997	yes it is all in the file name, and I love numbers 4 from back
Sun, 9 Dec 2012 15:53:49 +13:00	100003861284061	o..ok I think I got what you meant
Sun, 9 Dec 2012 15:54:35 +13:00	100003861284061	I assume it is last four from left to right?
Sun, 9 Dec 2012 15:55:43 +13:00	100003867343997	Yes
Sun, 9 Dec 2012 15:56:03 +13:00	100003861284061	ok all unique
Sun, 9 Dec 2012 15:56:14 +13:00	100003867343997	Yup
Sun, 9 Dec 2012 15:56:51 +13:00	100003861284061	great give me a second, wanna try it out just to make sure we got this right

UTC (Converted Local Time)	fbid Facebook ID	Chat Content
Sun, 9 Dec 2012 15:57:49 +13:00	100003867343997	ok if there is none to extract, it means none just keep going until you got one
Sun, 9 Dec 2012 15:59:41 +13:00	100003861284061	Ok
Sun, 9 Dec 2012 16:02:36 +13:00	100003861284061	ok I got it
Sun, 9 Dec 2012 16:03:42 +13:00	100003861284061	great! So same protocol in future and check for new post frequently in this melody group
Sun, 9 Dec 2012 16:03:53 +13:00	100003861284061	Ok
Sun, 9 Dec 2012 16:06:37 +13:00	100003867343997	oh one more thing
Sun, 9 Dec 2012 16:07:34 +13:00	100003867343997	just hit on the Like once you have read the message so that I know
Sun, 9 Dec 2012 16:07:51 +13:00	100003861284061	Ok

During the survey of the digital scene, steganography tools were identified by StegAlyzerAS. Based on the detection, both imaged hard drives were found to have significant file artefacts for two steganographic applications, JPHide v0.51 and JPSeek v0.51 (JP Hide and Seek) and Bon Kyu Bon v1.1.3011.2638. StegAlyzerAS results showed that 80% of JPHide and JPSeek v0.51 significant file artefacts were detected in Christian Riley's hard drive and 100% were found in John Doe's hard drive (Appendices 15 and 16). Although Bon Kyu Bon v1.1.3011.2638 was detected by StegAlyzerAS, it only showed a low percentage, 16.7% significant files detected on both suspects' hard drives (Appendix 17). Moreover, in Windows prefetch files, only JP Hide and Seek application (JPHSWIN.EXE) was found on both suspects' hard drives. Therefore, it is confirmed that JP Hide and Seek was the tool used by the suspects instead of Bon Kyu Bon v1.1.3011.2638. The execution files for JPHide and JPSeek v0.51, jphide.exe, was found to be located at I:\Users\Christian\Documents\Christian\jphs_05\jphs05\jphide.exe and R:\Users\John\Downloads\jphs_05\jphs05\jphide.exe respectively. It was determined that steganography was

involved based on the reconstructed Facebook chat and the artefacts that were found. Additionally, a downloaded URL for the JP Hide and Seek application, <http://ftp.gwdg.de>, was also found in the extracted data in John Doe's Internet history.

The Internet history extracted by Encase and IEF was further analyzed. It showed some Facebook file upload activity in the URL histories extracted. The relevant URL history looks as follows: http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=2&__iframe=true&__user=100003867343997. It was noticed that the Facebook user ID was located at the end of the link. This was the URL when the user uploaded a file onto the user's Facebook group. This URL was executed in between 3.14pm – 3.24pm on 09 Dec 2012 and between 12.11am – 12.18am on 10 Dec 2012. There were no indications of which files were uploaded in the URL history artefacts. However, the details of the uploaded file artefacts were identified in Encase with a keyword search for “uploaded a file” and the artefacts were found in pagefile.sys, blocks.mem, and nacl_irt_x86_32.nexe. These are the user's Facebook page fragments, an example is shown in Figure 4.13. The page fragment also contained Happy Farm (John Doe) file upload artefacts on Christian Riley's imaged hard drive. After eliminating duplicate file names, there were a total of six different file names of images uploaded by Christian Riley and four different file names uploaded by Happy Farm in Facebook. Additionally, the timestamps of these artefacts also matched the last accessed time of the extracted upload URL histories.

```
2191094;,&quot;type&quot;;1&#125;\">\u003Ca class=\"passiveName\" href=\"http://www.facebook.com/christia
2191195n.riley.3914\" data-ft=\"&#123;&quot;tn&quot;;&quot;;&quot;;&#125;\" data-hovercard=\"/\ajax/hovercar
2191296d\user.php?id=100003867343997\">Christian Riley\u003C/a> uploaded a file.\u003C/h5>\u003Cdiv class
2191397=\"mvm uiStreamAttachments fbMainStreamAttachment\" data-ft=\"&#123;&quot;type&quot;;10,&quot;tn&quot;
2191498;,&quot;H&quot;;&#125;\">\u003Cdiv class=\"clearfix\">\u003Cdiv class=\"clearfix uiAttachmentMediaFile
2191599s_8o_8t lfloat\" data-ft=\"&#123;&quot;type&quot;;40,&quot;tn&quot;;&quot;D&quot;;&#125;\" tabindex=
2191700\"-1\" aria-hidden=\"true\">\u003Ci class=\"_8o_8r lfloat img sp czc6sg sx 266747\">\u003C/i>\u003C
2191801div class=\"_8m_8u\">\u003Cspan class=\"fwb fcb\">2012-10-20 19-20-03 927.jpg\u003C/span>\u003Cdiv
2191902class=\"fileActionRow fsm fwn fcg\">\u003Ca class=\"uiLinkLightBlue\" href=\"#\" rel=\"dialog\" ajaxi
2192003fy=\"/\ajax/messaging/attachments/photo/dialog.php?uri=\u00252Fdownload\u00252F475114832531193\u0
21921040252F2012-10-20 19-20-03 927.jpg\" role=\"button\">Preview\u003C/a> \u00b7 \u003Ca class=\"uiLinkLig
2192205htBlue\" href=\"/download/475114832531193/2012-10-20 19-20-03 927.jpg\" rel=\"ignore\">Download\u0
219230603C/a> \u00b7 \u003Ca class=\"uiLinkLightBlue\" href=\"#\" rel=\"dialog\" ajaxify=\"/\ajax/groups/\
219240f/files/revision?message_id=179987902145576\" role=\"button\">Upload Revision\u003C/a>\u003Cdiv>\u0
219250803C/div>\u003Cdiv class=\"_8m_8u\">\u003Cdiv class=\"fsm fwn fcg\">\u003Cspan class=\"capt
2192609ion\" data-ft=\"&#123;&quot;tn&quot;;&quot;L&quot;;&#125;\">\u003Cspan>\u003Cdiv class=\"uiAttachmen
2192710tDesc translationEligibleUserAttachmentMessage\">\u003Cdiv>\u003Cdiv>\u003Cdiv>\u003Cdiv>\u00
21928113C/div>\u003Cform rel=\"async\" class=\"live_179987898812243_316526391751760 commentable_item autoex
2192912pand_mode\" method=\"post\" action=\"/\ajax/ufi/modify.php\" data-live=\"&#123;&quot;seq&quot;;0&#1
219301325;\" onsubmit=\"return window.Event &amp;&amp; Event.__inlineSubmit &amp;&amp; Event.__inlineSubmit(
2193114this,event)\" id=\"uri731e89\">\u003Cinput type=\"hidden\" name=\"charset_test\" value=\"%euro;,&acut
2193215e;,\u020ac,\u00b4,\u06c34,\u0414,\u0404\" /\u003Cinput type=\"hidden\" name=\"fb_dtsg\" value=\"AQOQi
2193316AWg\" autocomplete=\"off\" /\u003Cinput type=\"hidden\" autocomplete=\"off\" name=\"feedback_params
2193417\" value=\"&#123;&quot;actor&quot;;&quot;;100003867343997&quot;;&quot;;target_fb_id&quot;;&quot;;17998789
21935188812243&quot;;&quot;;target_profile_id&quot;;&quot;;&quot;;&quot;;type_id&quot;;&quot;;308&quot;;&quot;;as
2193619soc_obj_id&quot;;&quot;;&quot;;172888169522216&quot;;&quot;;source_app_id&quot;;&quot;;0&quot;;&quot;;extra_stor
2193720y_params&quot;;&quot;;&quot;;content_timestamp&quot;;&quot;;1355019452&quot;;&quot;;check_hash&quot;;&quot;;&quot;;A
2193821QDFYLSNP-FKA7TQ&quot;;&quot;;source&quot;;&quot;;0&quot;;&#125;\" /\u003Cinput type=\"hidden\" autocom
```

Figure 4.13: Facebook page fragment artefacts for file upload

The image file upload activities on Facebook were also found on John Doe imaged hard drive. The extracted URLs, http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=5&__iframe=true&__user=100003861284061 were executed in between 4.09pm and 4.19pm and at 5.02 pm on 09 Dec 2012 and in between 12.58am and 1.01am on 10 Dec 2012. Additional upload file name artefacts were found in pagefile.sys and an unallocated cluster of John Doe's imaged hard drive. After eliminating duplicate image file names, there were four image files uploaded onto Facebook by John Doe and six images uploaded by Christian Riley. The timestamps of the four images uploaded by John Doe matched four extracted Facebook upload URL histories.

There were also indications of image file downloads from Facebook on Christian Riley's imaged hard drive on 09 Dec 2012 at 3.25pm and between 5.09pm and 5.10pm and on 10 Dec 2012 at 12.21am, 12.40am and between 11.02am and 11.04am. The files downloaded showed a consistent URL file download pattern with a download ID [414901595250518], a file name [2012-09-21_21-13-51_504] and an extension [.jpg], for example, http://www.Facebook.com/download/414901595250518/2012-09-21_21-131_504.jpg. Another pattern for Facebook download URLs were also found. For example, http://attachment.fbsbx.com/file_download.php?id=414901595250518&eid=ASsDf_Gm0Xo4Rr4TvT_svACjRNngsy8vCbR2bdf-R0VCIVYEGtuDA01iwociJu5OMls&ext=1355026251&hash=ASvqIPWB-EkRZtnO found in the extracted data. The two URLs above are related as they have the same download ID number, **414901595250518**. So, when a user clicks on the download link to a shared file in Facebook, the first URL will be linked to the download domain and the download ID number relates to which file to download. It was also found that both URLs were executed within the same time frame. Both download URL patterns were found on John Doe's hard drive. The extracted URL history showed some image files were downloaded from Facebook at 4.22pm and 5.03pm on 09 Dec 2012, and at 1.02am and between 1.23am and 1.28am on 10 Dec 2012.

Furthermore, the upload and download artefacts extracted from both suspects' machines were looked at, it was found that the file name of six image files that Christian Riley had uploaded on Facebook could be found on the

Facebook download URL history extracted from John Doe's machine. Likewise, the four file names uploaded by John Doe could be found in Christian Riley's Facebook download URL history. However, an investigator needs to be aware that having the same file name does not prove that those files are exact copies.

The image file name pattern similar to *2012-09-21_21-13-51_504.jpg* were then looked up. These file name patterns were identified in a few locations on the suspects' machines as shown in Table 4.10. There were a total of 18 similar image file name patterns found on Christian Riley's imaged hard drive and 14 of them had the same file name as that found in the extracted Facebook download URL history. The same process was undertaken on John Doe's imaged hard drive, 16 similar image file name patterns were found and 12 of them had the same file name as that found in the extracted Facebook download URL history. All of these image files appear to be regular image files, but, it is evident that there are some changes to these files as some same image files saved in different locations of the computer have different MD5 values. Thus, the image files that had the same name in the download URL history were exported for further analysis.

Table 4.10 Identified Image File Locations

Christian Riley's image hard drive:	John Doe's image hard drive:
C:\Users\Christian\Downloads\	C:\Users\John\Downloads\
C:\Users\Christian\Pictures\from John\	C:\Users\John\Downloads\special photos\
C:\Users\Christian\Pictures\Photos\	C:\Users\John\Pictures\from Christ\
C:\Users\Christian\Pictures\Special pictures\	C:\Users\John\Pictures\Photos\
	C:\Users\John\Pictures\To Christ\

According to the results from StegAlyzerSS, one file similar to the download file name, *2012_12-01_16-42-35_679.jpg*, was identified as having appended data on both suspects' imaged hard drives. When processing the identified image file in StegAlyzerSS, the appended data was not readable, so by searching the signature HEX value, FFD9, of the image's end of file (EOF), it was found that a .rar file header signature (HEX value 52 61 72 21 or Rar!) was found right after the EOF of the identified image. An image file named *u.jpg* was identified (Figure 4.14). This is meant that *u.jpg* was compressed in a .rar file and appended to the

identified image file. Both u.jpg and u.rar files were found in Christian Riley's imaged hard drive, but, could not be identified on John Doe's imaged hard drive. As it was discovered that 2012_12-01_16-42-35_679.jpg was appended with a .rar file, the 2012_12-01_16-42-35_679.jpg found on John Doe's imaged hard drive was exported and opened with WinRaR application as well. This execution revealed the appended u.jpg file.

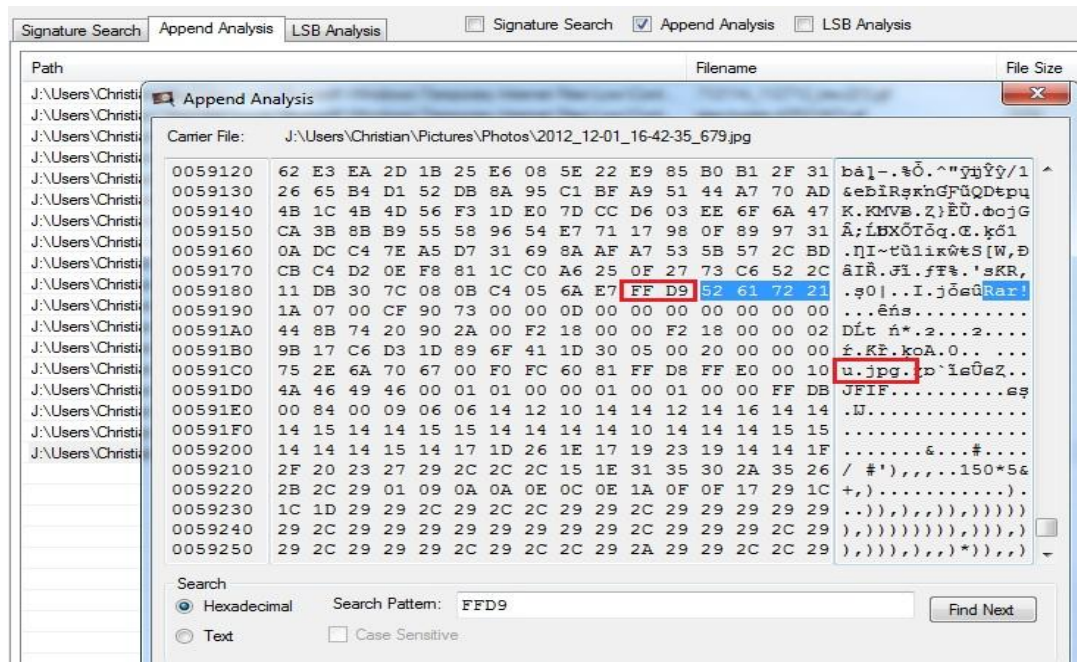
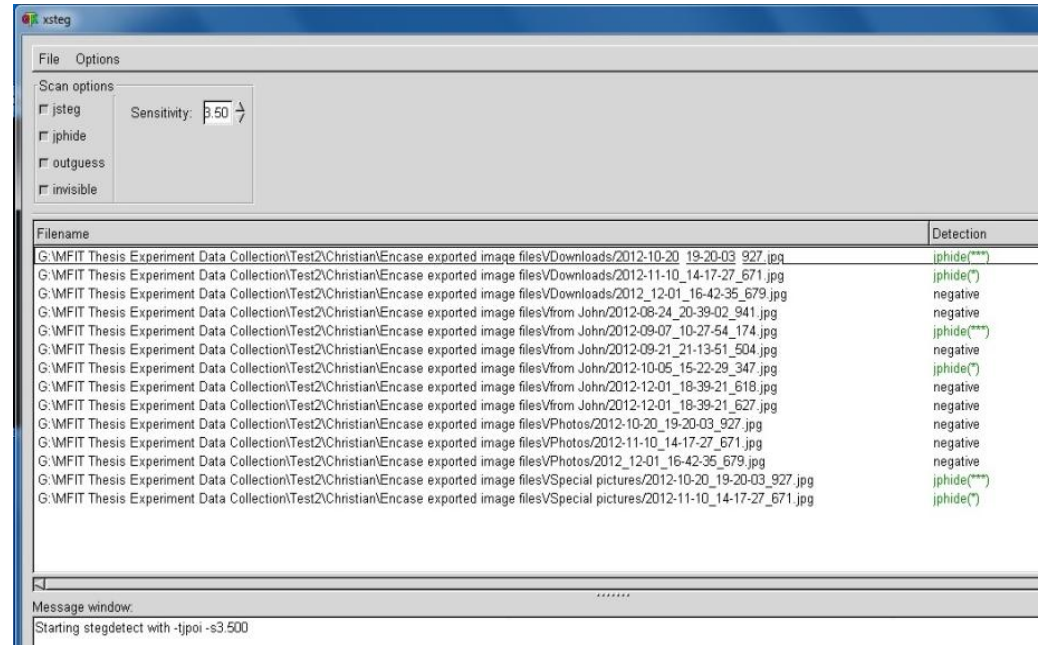


Figure 4.14: Identified image file with appended data

Since StegAlyzerSS had failed to detect any JP Hide and Seek signatures in suspects' imaged hard drives, StegDetect Windows version which was known to be able to detect JP Hide and Seek's signature within images was executed for detection. The sensitivity value of StegDetect was set at 3.5. The higher the value, the more sensitive it is in detecting any discrepancies in an image. The results showed that six images on Christian Riley's imaged hard drive and seven images on John Doe's imaged hard drive were identified as having JP Hide and Seek signatures. As seen in Figures 4.15 and 4.16, some images were detected as having three green asterisks (***) and some only one asterisk (*). The numbers of asterisks represent how significant the statistical signature of that particular steganographic algorithm is. Three is the highest and one is the lowest.

Referring to the Facebook chat recovered previously, it was understood that not all images had been embedded with secret messages by the suspects and

the passphrase for extracting the secret message was the last four digits from left to right of the file name. So, each image file detected as containing JP Hide and Seek's signature was opened one-by-one with the JP Hide and Seek application and extracted with the passphrase pattern learnt from the recovered Facebook chat in the investigator machine. This process successfully extracted six secret messages from all images identified on Christian Riley's imaged hard drive and six secret messages from the seven images identified on John Doe's imaged hard drive. There was one image identified with a low significant statistical signature located at D\Users\John\Pictures\Photos\2012-09-07_10-27-54_174.jpg with an MD5 value of 5ac2497d7a3359070dcea457a658e436 on John Doe's imaged hard drive that was unable to be extracted and showed an error message; "wrong passphrase". There are two possibilities for this error. It is either that there was no secret message to extract or it was indeed the wrong passphrase. As the other three images with the same file name, had higher significant steganographic signatures, had the same MD5 values and their embedded secret messages could be successfully extracted, this particular image, having different MD5 values and a low significant signature was probably a false positive image.



Filename	Detection
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\Downloads\2012-10-20_19-20-03_927.jpg	jphide(***)
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\Downloads\2012-11-10_14-17-27_671.jpg	jphide(*)
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\Downloads\2012_12-01_16-42-35_679.jpg	negative
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\from John\2012-08-24_20-39-02_941.jpg	negative
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\from John\2012-09-07_10-27-54_174.jpg	jphide(***)
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\from John\2012-09-21_21-13-51_504.jpg	negative
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\from John\2012-10-05_15-22-29_347.jpg	jphide(*)
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\from John\2012-12-01_18-39-21_618.jpg	negative
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\from John\2012-12-01_18-39-21_627.jpg	negative
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\Photos\2012-10-20_19-20-03_927.jpg	negative
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\Photos\2012-11-10_14-17-27_671.jpg	negative
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\Photos\2012_12-01_16-42-35_679.jpg	negative
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\Special pictures\2012-10-20_19-20-03_927.jpg	jphide(***)
G:\MFIT Thesis Experiment Data Collection\Test2\Christian\Encase exported image files\Special pictures\2012-11-10_14-17-27_671.jpg	jphide(*)

Message window:
Starting stegdetect with -tjpoi -s3.500

Figure 4.15: Steganographic images detected by StegDetect on Christian Riley's imaged hard drive

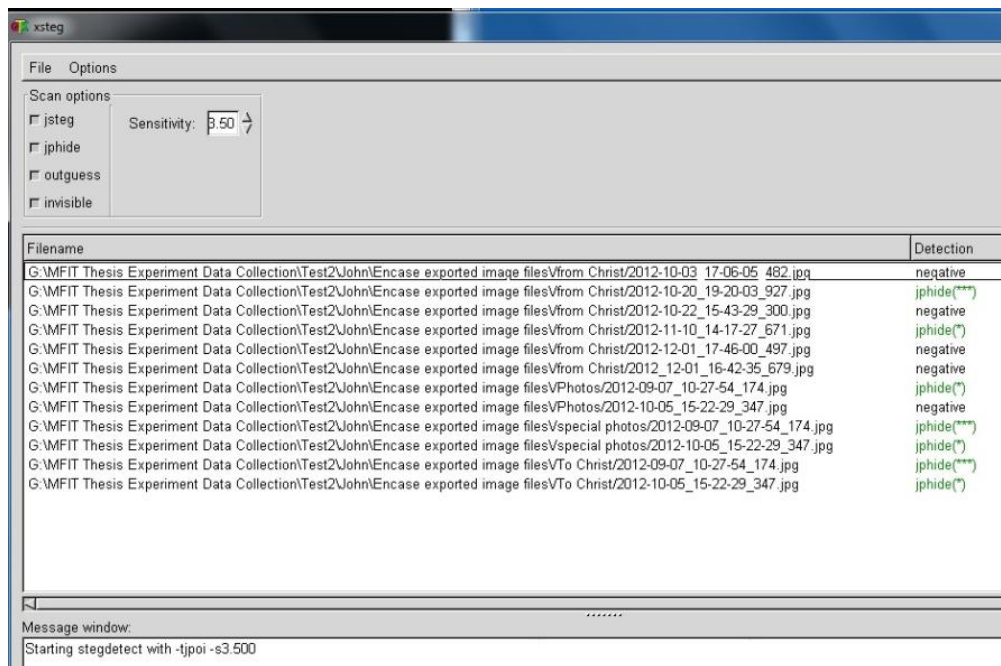


Figure 4.16: Steganographic images detected by StegDetect on John Doe's imaged hard drive

Previously, the software was unable to prove that images with the same file names found on both suspects' machines, file upload artefacts and Facebook download URLs were exact copies. However, by comparing the MD5 values of all the image of interest found on both suspects' imaged hard drives, the MD5 values of the suspect image files were the same, confirming that these files are exact copies. Furthermore, all these matching image files not only had the same MD5 values but also had matching file names in the extracted Facebook download URL history and the Facebook file upload artefacts found in pagefile.sys, Blocks.mem and unallocated clusters.

Table 4.11: Timeline Analysis

Suspect Name	Artefacts	First accessed date	First accessed time	Last accessed date	Last accessed time
Christian Riley	Jphswin.exe	9/12/2012	15:14:53	10/12/2012	11:04:42
Christian Riley	Facebook download URL history	9/12/2012	15:26:42	10/12/2012	11:04:59
John Doe	Jphswin.exe	9/12/2012	16:00:08	10/12/2012	01:29:16
John Doe	Facebook download URL history	9/12/2012	15:33:03	10/12/2012	01:29:02

Furthermore, when comparing the timeline of the last execution of the JP Hide and Seek (JPHSWIN.EXE) with the last downloaded file from the Facebook download URL history, the timeframes match (Table 4.11).

In addition, .txt files were also looked up on both suspects’ imaged hard drive, like the one shown in Figure 4.17. It is well known that .txt is the only file type that can be embedded or extracted as a secret message in JP Hide and Seek. Therefore, it is worthwhile looking at this possible file type. By tracing the .txt file to its location there were four files of interest that had suspicious content found on Christian Riley’s imaged hard drive; newID.txt, Mission.txt, fromJohn.txt, and fromjohn1.txt. Likewise, four suspicious files of interest were found on John Doe’s imaged hard drive; christ1.txt, christ2.txt, address.txt, and To HIM.txt. When opened, the content of these text files was shown to be the same as that of the extracted secret messages.

	Target	Document
<input type="checkbox"/> 9	Terrorism Related Case	C:\Users\John\Documents\To HIM.txt
<input type="checkbox"/> 10	Terrorism Related Case	C:\Users\John\Documents\To HIM.txt
<input type="checkbox"/> 11	Terrorism Related Case	C:\Users\John\Downloads\christ1.txt
<input type="checkbox"/> 12	Memory Dump	C:\Users\John\Downloads\christ1.txt
<input type="checkbox"/> 13	Memory Dump	C:\Users\John\Downloads\christ1.txt
<input type="checkbox"/> 14	Terrorism Related Case	C:\Users\John\Downloads\christ1.txt
<input type="checkbox"/> 15	Terrorism Related Case	C:\Users\John\Downloads\christ1.txt
<input type="checkbox"/> 16	Memory Dump	C:\Users\John\Downloads\christ1.txt
<input type="checkbox"/> 17	Terrorism Related Case	C:\Users\John\Downloads\special photos\tochrist.txt
<input type="checkbox"/> 18	Memory Dump	C:\Users\John\Downloads\special photos\tochrist.txt
<input type="checkbox"/> 19	Memory Dump	C:\Users\John\Downloads\special photos\tochrist.txt
<input type="checkbox"/> 20	Terrorism Related Case	C:\Users\John\Downloads\special photos\tochrist.txt
<input type="checkbox"/> 21	Terrorism Related Case	C:\Users\John\Downloads\special photos\tochrist2.txt
<input type="checkbox"/> 22	Terrorism Related Case	C:\Users\John\Downloads\special photos\tochrist2.txt

Figure 4.17 .txt files extracted by Encase (John Doe)

4.3.2.6 Conclusion

Based on the extracted and reconstructed data, it is evident that both suspects were communicating in Facebook chat and sharing steganographic image files on Facebook. Both suspects’ Facebook IDs and group ID were identified on both suspects’ machines. The Facebook chats performed by both users were found on both machines also. The timeline of JPHSWIN.EXE execution and the Facebook

file download history matched. Both suspects' hard drives were also found to have matching MD5 values for the steganographic images identified. This proves that both suspects have exact copies of the image files. Furthermore, the steganographic image file names identified matched the file names found in the Facebook download URL history and partly matched the upload artefacts in Facebook page fragments. Finally, the secret messages embedded in the steganographic images identified were all successfully extracted with the help of the extracted Facebook chat.

4.3.3 Comparative Analysis

The objective of this comparison is to compare the control data and the reconstructed data after forensic analysis. On the far left of Table 4.12 is the control data performed during the simulation of Case Scenario 1 and the right of the table indicates whether the known artefacts in the control data were found in the reconstructed data during the digital forensics analysis and also indicates how the evidence was found.

Table 4.12: Scenario 1 Comparative Analysis

Control Data	Reconstructed Data	
(Known Artefacts)	(Evidence)	(How)
Steganography Tool – JP Hide and Seek (JPHSWIN.EXE)	Found	Detected by StegAlyzerAS, execution artefacts found in Windows prefetch files
Steganographic Image - 2012-10-20_19-20-03_927.jpg (JP Hide and Seek)	Found	Lead by Facebook history download URLs and the steganographic signature was detected by StegDetect
Steganographic Image - 2012_12-01_16-42-35_679.jpg (EOF Injection)	Found	Detected by StegAlyzerSS
Steganographic Image - 2012-11-10_14-17-27_671.jpg (JP Hide and Seek)	Found	Lead by Facebook history download URLs and the steganographic signature was detected by StegDetect
Steganographic Image - 2012-09-07_10-27-54_174.jpg (JP Hide and Seek)	Found	Lead by Facebook history download URLs and the steganographic signature was detected by StegDetect

Control Data	Reconstructed Data	
(Known Artefacts)	(Evidence)	(How)
Steganographic Image - 2012-10-05_15-22-29_347.jpg (JP Hide and Seek)	Found	Lead by Facebook history download URLs and the steganographic signature was detected by StegDetect
Secret Messages	Extracted	With a hint from Facebook chat
Facebook Files Upload	Found	EnCase Internet Artefacts Search
Facebook Files Download	Found	EnCase Internet Artefacts Search
Facebook Chat	Found	EnCase keyword search

In Case Scenario 1, the steganographic tool's artefacts were identified by StegAlyzerAS and its execution artefacts were then found in the Windows prefetch file. All the steganographic images were successfully identified by StegDetect with a hint from the internet history artefacts and the secret messages were successfully extracted with help from the Facebook chat artefacts. It was learned from the experimental case scenario that the automated steganalysis tool – StegDetect was able to detect the JPHide's signature whereas StegAlyzerSS was not capable of identifying such a signature. Additionally, the artefacts from internet history and the social network itself played a significant role in leading the investigator to look for steganographic images that resided in the suspect's machine and were associated with the online social network.

4.4 CASE SCENARIO 2 – INTELLECTUAL PROPERTY

The second scenario is about a disgruntled member of staff that was sending a company's intellectual property to a competitor using image steganography in Google+. The objective of the investigation is to identify, extract, and analyse any potential evidence on John's work station associated with distributing confidential company information from his Google+ account. The expected outcome is to identify a relationship between Starword's confidential documents and steganographic images. It is expected that the executed steganographic tool's

artefacts, the created steganographic images, and artefacts that show that steganographic images were in the user's Google+ content can be identified and it is also expected that the secret messages can be recovered.

4.4.1 Environment Set Up

Scenario 2 used a laptop equipped with Wifi connection, Intel ® Core TM 2 Duo CPU, 2.20GHz, 2GB RAM and 160GB hard drive (Target Machine 3 – John Doe) and the photos used in the simulation were captured with a Canon IXUS digital camera, with all pictures resized to 640 X 480 pixels. The hard drive was fully wiped with DBAN and freshly installed with Windows 7 Professional Edition.

For the forensic investigation environment, the data collection of the target's hard drive was performed with a software write blocker called SAFE Block Win 7 and FTK Imager 3.0. The data acquisition setup was similar to Scenario 1 as depicted in Figure 4.9 and all the acquired evidence image files were verified with MD5 and SHA hash values and saved as Encase evidence files (.E01) in an external 1TB hard drive.

4.4.2 Digital Forensics

The digital forensic process for Scenario 2 also adopted the digital forensic phases proposed by Noureldin, Hashem, and Abdalla (2011). The process steps are: 1) Evaluation and Assessment 2) Acquisition of Digital Evidence 3) Survey of Digital Scene (optional) 4) Digital Evidence Examination 5) Reconstruction of Extracted Data 6) Conclusion.

4.4.2.1 Evaluation and Assessment

- Laptop was powered off when seized.
- Only suspect's hard drive was sent to the forensics lab
- Tools needed: SATA to USB connector, software write blocker SAFE Block Win7, FTK Imager 3.0, Encase 7.0, Internet Evidence Finder, WinPrefetchView, StegAlyzerAS, StegAlyzerSS

4.4.2.2 Acquisition of Digital Evidence

- It is a Western Digital hard drive
- Model: WD1600BEVS
- Storage: 160GB

- Serial Number: WXEZ07A58058

The suspect's hard drive was connected to the investigator machine with a SATA to USB connector. The investigator machine, installed with SAFE Block Win 7 software write blocker and FTK imager, was used to image the suspect's hard drive bit-by-bit and saved it on an external hard drive as S2.E01. The integrity of the S2.E01 file was verified with MD5 and SHA hash values. After acquisition the physical hard drive was kept in a secure place.

4.4.2.3 Survey of Digital Scene

The suspect's imaged hard drive was mounted in StegAlyzerAS and StegAlyzerSS to search for steganographic tool artefacts and steganographic image artefacts. The evaluation of the imaged hard drive found three applications containing unique steganographic file artefacts, five applications containing detected registry artefacts, 0 signature files, four appended image files, and four files having LSB embedding (Table 4.13).

Table 4.13: StegAlyzerAS and StegAlyzerSS Detection Summary (Scenario 2)

Forensic Tool	Steganographic Artefacts Detected	No. of Applications Found
		S2.E01
StegAlyzerAS	Unique File Artefacts	3
StegAlyzerSS – Registry Artefacts	Registry Artefacts	5
StegAlyzerSS - Signature Analysis	Signature Artefacts	0
StegAlyzerSS - Append Analysis	Appended Artefacts	4
StegAlyzerSS -LSB Analysis	LSB Artefacts	4

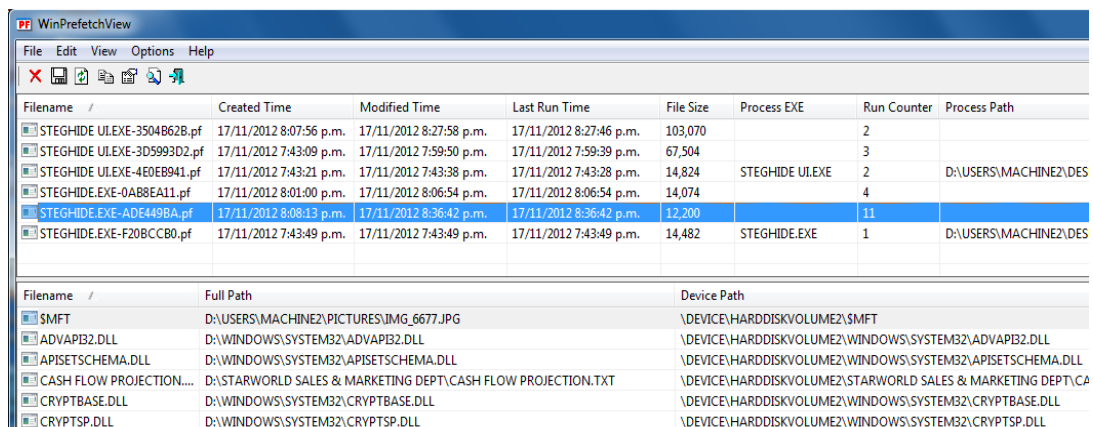
4.4.2.4 Digital Evidence Examination

The imaged hard drive evidence file was added into Encase 7.0 for data extraction and evidence processing. Each file in the evidence file was hashed with MD5 and SHA to ensure the integrity of the data files. Internet artefacts were also automatically extracted. Internet Evidence Finder was also used to extract internet activities. Table 4.14 is a summary of the data that was extracted from the target's hard drive.

Table 4.14 Google + Internet history data extracted

Forensic Tool	Domain	No. URLs Visited	Total Visits
Encase v7	plus.google.com/	27	241
	account.google.com/	24	73
	google.co.nz	37	254
	google.co.nz/	42	150
IEF v5	plus.google.com/	17	224
	account.google.com/	14	49
	google.co.nz	9	32
	google.co.nz/	23	77

Detection of steganographic artefacts in the previous stage has encouraged the investigator to look into the traces of steganographic application execution in the Windows prefetch files. Windows prefetch data was previewed with EnCase and it was found that there were six prefetch files related to StegHide application but no prefetch files were associated with the other two applications indicated in StegalyzerAS. Therefore, only StegHide was the focus for further investigation. The earliest created time for a StegHide prefetch file was on 17/11/12 07:43:09pm and the last written time was on 17/11/12 08:36:42pm on the suspect's machine. These six prefetch files were then exported for further extraction to WinPrefetchView. STEGHIDE.EXE-ADE449BA.pf showed to have the highest run count of 11 times, created on 17/11/12 08:08:13pm and last run on 17/11/12 08:36:42pm (Figure 4.18).



Filename	Created Time	Modified Time	Last Run Time	File Size	Process EXE	Run Counter	Process Path
STEGHIDE.ULEXE-3504862B.pf	17/11/2012 8:07:56 p.m.	17/11/2012 8:27:58 p.m.	17/11/2012 8:27:46 p.m.	103,070		2	
STEGHIDE.ULEXE-3D5993D2.pf	17/11/2012 7:43:09 p.m.	17/11/2012 7:59:50 p.m.	17/11/2012 7:59:39 p.m.	67,504		3	
STEGHIDE.ULEXE-4E0EB941.pf	17/11/2012 7:43:21 p.m.	17/11/2012 7:43:38 p.m.	17/11/2012 7:43:28 p.m.	14,824	STEGHIDE.ULEXE	2	D:\USERS\MACHINE2\DES
STEGHIDE.EXE-0AB8EA11.pf	17/11/2012 8:01:00 p.m.	17/11/2012 8:06:54 p.m.	17/11/2012 8:06:54 p.m.	14,074		4	
STEGHIDE.EXE-ADE449BA.pf	17/11/2012 8:08:13 p.m.	17/11/2012 8:36:42 p.m.	17/11/2012 8:36:42 p.m.	12,200		11	
STEGHIDE.EXE-F208CCB0.pf	17/11/2012 7:43:49 p.m.	17/11/2012 7:43:49 p.m.	17/11/2012 7:43:49 p.m.	14,482	STEGHIDE.EXE	1	D:\USERS\MACHINE2\DES

Filename	Full Path	Device Path
SMFT	D:\USERS\MACHINE2\PICTURES\IMG_6677.JPG	\DEVICE\HARDDISKVOLUME2\SMFT
ADVAPI32.DLL	D:\WINDOWS\SYSTEM32\ADVAPI32.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\ADVAPI32.DLL
APISETSCHEMA.DLL	D:\WINDOWS\SYSTEM32\APISETSCHEMA.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\APISETSCHEMA.DLL
CASH FLOW PROJECTION...	D:\STARWORLD SALES & MARKETING DEPT\CASH FLOW PROJECTION.TXT	\DEVICE\HARDDISKVOLUME2\STARWORLD SALES & MARKETING DEPT\CA
CRYPTBASE.DLL	D:\WINDOWS\SYSTEM32\CRYPTBASE.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\CRYPTBASE.DLL
CRYPTSP.DLL	D:\WINDOWS\SYSTEM32\CRYPTSP.DLL	\DEVICE\HARDDISKVOLUME2\WINDOWS\SYSTEM32\CRYPTSP.DLL

Figure 4.18: Most active STEGHIDE.EXE prefetch file that found on John Doe's machine

4.4.2.5 Reconstruction of Extracted Data

Further analysis was carried out on these prefetch files in WinPrefetchView. By looking at the content of each prefetch file, it was found that STEGHIDE.EXE-ADE449BA.pf and STEGHIDE.EXE-0AB8EA11.pf contained files associated with images located in \DEVICE\HARDDISKVOLUME2\USERS \MACHINE2 \PICTURES\ and text documents located in \DEVICE\HARDDISK VOLUME2\STARWORLD SALES & MARKETING DEPT\ (Figure 4.19). This implies that STEGHIDE.EXE was assessing these files during its execution. Furthermore, these text files are confidential documents belonging to Starworld. Consequently, the directory, USERS\MACHINE2 \PICTURES\ was then traced and seven image files were found. Five of these image files were files indicated in both prefetch files. These are the highly suspect image files that could be the steganographic images. As StegAlyzerSS did not identify StegHide's signature in Section 4.4.2.3, HEX value analysis was conducted on these suspect images to look for any unusual patterns in these image files. It was found that each of the image files had an unusual persistent HEX value pattern after the header file similar to Figure 4.20. This confirmed that the image had been manipulated as a regular, clean digital image (Figure 4.21) will not have such a signature.



Filename	Full Path
\$MFT	D:\USERS\MACHINE2\PICTURES\IMG_6677.JPG
ADVAPI32.DLL	D:\WINDOWS\SYSTEM32\ADVAPI32.DLL
APISETSCHEMA.DLL	D:\WINDOWS\SYSTEM32\APISETSCHEMA.DLL
CASH FLOW PROJECTION.TXT	D:\STARWORLD SALES & MARKETING DEPT\CASH FLOW PROJECTION.TXT
CRYPTBASE.DLL	D:\WINDOWS\SYSTEM32\CRYPTBASE.DLL
CRYPTSP.DLL	D:\WINDOWS\SYSTEM32\CRYPTSP.DLL
GDB2.DLL	D:\WINDOWS\SYSTEM32\GDB2.DLL
IMG_2255.JPG	D:\USERS\MACHINE2\PICTURES\IMG_2255.JPG
IMG_2292.JPG	D:\USERS\MACHINE2\PICTURES\IMG_2292.JPG
IMG_6677.JPG	D:\USERS\MACHINE2\PICTURES\IMG_6677.JPG
IMM32.DLL	D:\WINDOWS\SYSTEM32\IMM32.DLL
KERNEL32.DLL	D:\WINDOWS\SYSTEM32\KERNEL32.DLL
KERNELBASE.DLL	D:\WINDOWS\SYSTEM32\KERNELBASE.DLL
LOCALE.NLS	D:\WINDOWS\SYSTEM32\LOCALE.NLS
LPK.DLL	D:\WINDOWS\SYSTEM32\LPK.DLL
MSCTF.DLL	D:\WINDOWS\SYSTEM32\MSCTF.DLL
MSVCRT.DLL	D:\WINDOWS\SYSTEM32\MSVCRT.DLL
NOTE.TXT	D:\USERS\MACHINE2\DESKTOP\NOTE.TXT
NOTE2.TXT	D:\USERS\MACHINE2\DESKTOP\NOTE2.TXT
NTDLL.DLL	D:\WINDOWS\SYSTEM32\NTDLL.DLL
PROMO02_12.TXT	D:\STARWORLD SALES & MARKETING DEPT\PROMO02_12.TXT
PSAPI.DLL	D:\WINDOWS\SYSTEM32\PSAPI.DLL

Figure 4.19: Files contained in STEGHIDE.EXE-0AB8EA11.pf

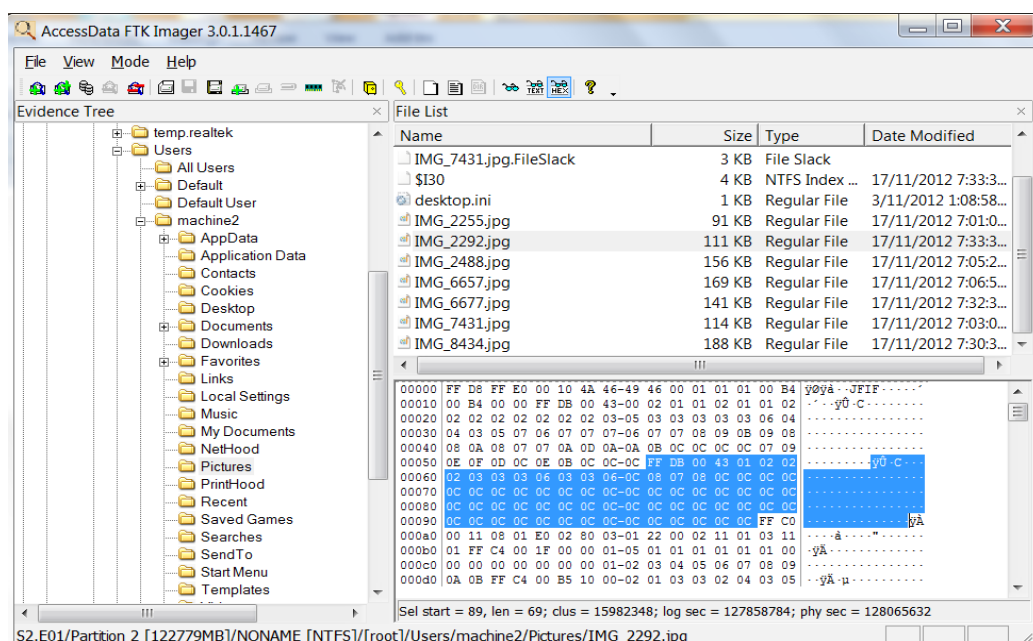


Figure 4.20: HEX value in the header of the suspect steganographic image

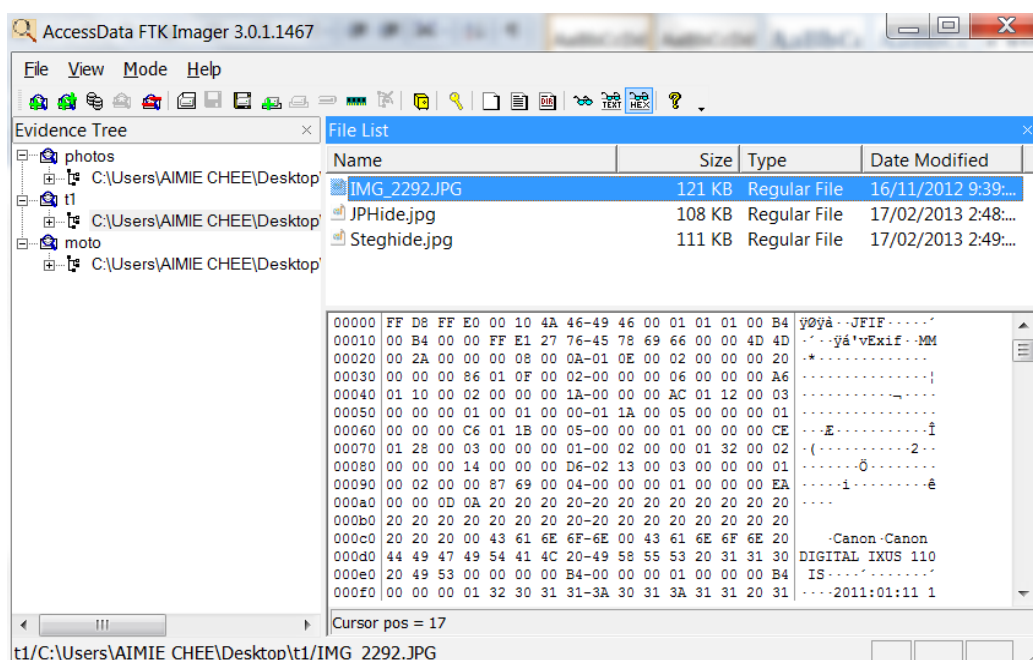


Figure 4.21: HEX value in the header of a regular, clean digital image

Subsequent analysis was then carried out to determine the suspect's Google+ activities. It is evident that Google+ had been actively accessed by the suspect with the second highest visit count of 241 for its domain, plus.google.com. The photo upload URL associated with Google+ was also identified by looking at the extracted data related to the plus.google.com domain. The photo upload URL in Google+ is similar to https://plus.google.com/_upload/photos/resuma

ble?authuser=0&upload_id=AEnB2Uo7irt8P_UgITbw4ucT6eQeaDqnCY0i4ffV-mZsxsjxket92wDAP9k0RpApQ-SYyzQdhubEzKzvW_NWMzhl0NxCBMKdwBQ
&file_id=000. It was noted that the upload ID in the URL is encrypted by Google+, and therefore unable to be read seven different encrypted photo upload IDs were identified and the last accessed times ranged from 17/11/12 08:09:29pm to 17/11/12 08:11:11pm and from 17/11/12 08:37:39pm to 17/11/12 08:38:51pm. The timeframe of photo upload and active usage of STEGHIDE.EXE overlapped; therefore it is highly likely that these photos uploaded onto Google+ may have been embedded with confidential text documents as revealed previously in the prefetch files, STEGHIDE.EXE-ADE449BA.pf and STEGHIDE.EXE-0AB8EA11.pf.

As it was not possible to tell from the extracted photo upload URL which photos were uploaded, the images cached in the browser cache files were then looked up in EnCase. There were 516 pictures extracted from the browser cache file and each identified picture had the specific URL name it belonged to. With the help of EnCase picture gallery, the seven suspicious image files previously identified, having a file name pattern of IMG_[number].jpg in USERS\MACHINE2 \PICTURES\ were found in browser cache files with a URL name similar to https://lh[number].googleusercontent.com/.../.../.../.../.../[image file name], for example https://lh3.googleusercontent.com/-IiLoZI47Gio/UKc-wm-BBSI/AAAAAAKw/yLkIR54p_rU/w497-h373/IMG_6677.jpg. This is the URL pattern for them the browser downloads a particular image file from the user's Google+ content and saves it as a temporarily copy in the browser cache file. Therefore, by looking at this pattern of URL, 32 images were identified and these images displayed the same images as the seven suspect steganographic images found in USERS\MACHINE2\PICTURES\ (Figure 4.22). This coincidence is highly suspicious.

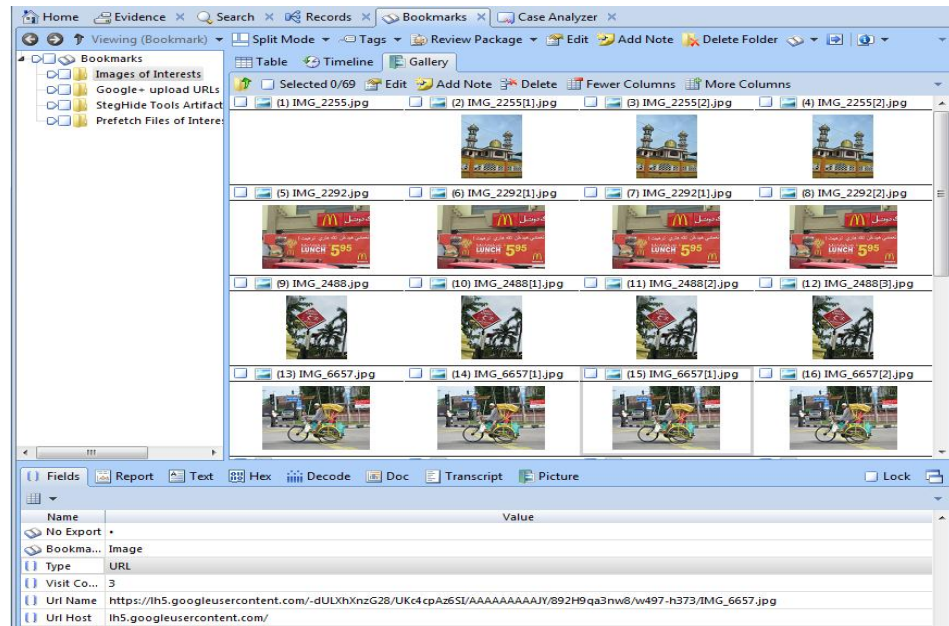


Figure 4.22: Displayed images of interest found in the browser cache files extracted by EnCase software.

In addition, visited link history records extracted from Internet Explorer (Windows) in EnCase also showed that the confidential documents from Starword Sales & Marketing department had been actively accessed on 17/11/12 between 08:00:19pm and 08:33:17pm. The 7 image files that found to be suspicious were also been actively accessed on 17/11/12 in between 08:00:50pm and 08:36:27pm. Seven additional image files that had the same file name as the suspicious image files were also found in an additional path; *E:/John%20Doe/photos/* and had been accessed on the same day between 08:00:04pm and 08:33:01pm. This drive *E:/* is not the primary drive found in the suspect's hard drive, but is likely to be a path for an external drive. The reconstruction timeframe of these visited links showed that some of these files were accessed in a sequential pattern based on the last accessed time of each file. An image file was first accessed from *E:/John Doe/photos/*, then a text file document in *C:/StarWord Sales & Marketing Dept/* and finally an image file in *C:/Users/machine2/Pictues/*. The artefact of this pattern is depicted in Appendix 27. Overall, the timeframe of these consecutive activities are within the timeline of the steghide.exe execution as well as the Google+ photo upload history that fall between 8.00 pm and 8.38pm on 17 Nov 2012 (Figure 4.23).

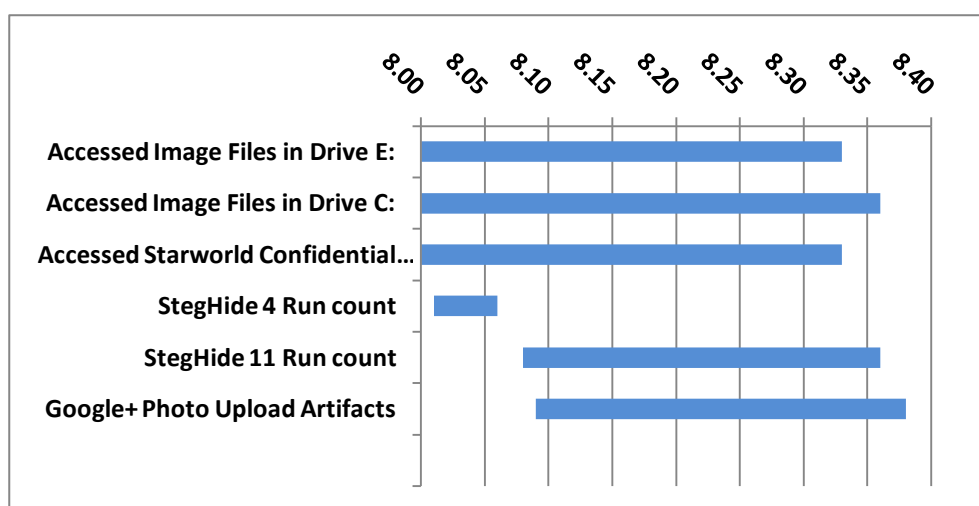


Figure 4.23: Suspicious activities timeline

A keyword search performed by EnCase also revealed some interesting information about the suspect's Google + content. The suspect's user name and email address in Google+ was identified as John Doe and happyfarm0921@gmail.com respectively. Additionally, the suspect's Google+ ID number was identified as 111267948980380534594. A suspicious post messages in Google+ between the suspect and XO Mart's managing director, Christian Riley was identified. This message was posted on 17/11/2012 between 8.11pm and 8.17pm (Table 4.15). Additionally, a Google+ page fragment was found containing one URL image, https://lh4.Googleusercontent.com/-B2GUvIG0UIY/UKc4jsUCSsI/AAAAAAAAAJw/0x7ZWTXZeaw/IMG_2488.jpg with a suspect message "*you deserve this! haha...*" posted on 17/11/2012 at 8.11m.

Table 4.15: Reconstructed Message Posted

Name	Message	Unix Time	Converted Local Time
John Doe	when are you free for a coffee?	1353136318	Sat, 17 Nov 2012 20:11:58 +13:00
Christian Riley	c u! remember to bring the tool to show me!	1353136421	Sat, 17 Nov 2012 20:13:41 +13:00
John Doe	okie dokie!	1353136658	Sat, 17 Nov 2012 20:17:38 +13:00

4.4.2.6 Conclusion

Based on the extracted and reconstructed data, it is evident that the suspect was actively accessing Starworld's confidential sales and marketing department's data and was active on Google+ at a similar timeframe. StegHide.exe artefacts identified by StegAlyzerAS were also ascertained by execution artefacts and associated text files and image files left in the Windows pretetch files. Furthermore, the timeframe of the StegHide.exe executions fell into the last accessed time of various artefacts ranging from Starworld's confidential text documents, suspected image files, and Google+ upload history. These identified artefacts have positively shown that Starworld's confidential documents have been embedded by the suspect into the suspected image files and distributed via Google+ photo upload. The unusual pattern of HEX values is also another indication of image data manipulation on the suspected images. The suspect images displayed the same image as the ones identified in the browser cache file. Additionally, suspicious messages were found in the suspect's Google+ content including interaction with Starworld's managing director, Christian Riley, have underscored the suspect's abnormal activities. However, at this stage there is no direct evidence to prove that the seven suspected image files found on the suspect's work station were indeed embedded with Starworld's confidential documents as there was no indication of a passphrase to extract the secret message from these suspect images.

To prove that Starworld's confidential documents were embedded in these seven suspect image files, additional cryptanalysis, consent and interview to get the passphrase from the suspect will be needed. Once the passphrase is given by the suspect, each of the suspicious image files can be exported and the embedded file can be extracted by the StegHide application to reveal the content of the embedded file.

4.4.3 Comparative Analysis

Table 4.16 is a comparison table between the control data of Case Scenario 2 during the simulation and the reconstructed data collected during the digital forensics analysis. In Case Scenario 2, steganographic tool artefacts were identified by StegAlyzerAS and its execution artefacts associated with the text

files and image files were found in the Windows prefetch files as well. This information found in the prefetch files certainly aided the forensic examination. However, the steganalysis tool used in the experiment – StegAlyzerSS was unable to identify the steganographic signature of the suspected images in Case Scenario 2 and StegDetect was not designed for detecting StegHide’s signature. Therefore, even though there were other probative artefacts that called into question the content of the identified images; the identified images themselves were unable to prove the existence of steganographic content unless a further steganalytic process is able to prove the existence of the secret message. Thus, in this comparative analysis, images identified during the forensic analysis are only designated suspected rather than found, as the steganographic signature cannot be identified by the steganalysis tool available and the passphrase needed to extract the secret message is not available (Table 4.16).

Table 4.16: Scenario 2 Comparative Analysis

Control Data - Known Artefacts		Reconstructed Data	
	Evidence	How	
Steganography Tool – StegHide	found	Detected by StegAlyzerAS, execution artefacts found in Windows prefetch files	
Steganographic Image - IMG_2255.jpg (StegHide)	suspected	Lead by information in Windows Prefetch Files, browser cache images, HEX value, and recent active files accessed	
Steganographic Image - IMG_7431.jpg (StegHide)	suspected	Lead by information in Windows Prefetch Files, browser cache images, HEX value, and recent active files accessed	
Steganographic Image - IMG_6657.jpg (StegHide)	suspected	Lead by information in Windows Prefetch Files, browser cache images, HEX value, and recent active files accessed	
Steganographic Image - IMG_2488.jpg (StegHide)	suspected	Lead by information in Windows Prefetch Files, browser cache images, HEX value, and recent active files accessed	

Control Data - Known Artefacts	Reconstructed Data	
	Evidence	How
Steganographic Image - IMG_6677.jpg (StegHide)	suspected	Lead by information in Windows Prefetch Files, browser cache images, HEX value, and recent active files accessed
Steganographic Image - IMG_8434.jpg (StegHide)	suspected	Lead by information in Windows Prefetch Files, browser cache images, HEX value, and recent active files accessed
Steganographic Image - IMG_2292.jpg (StegHide)	suspected	Lead by information in Windows Prefetch Files, browser cache images, HEX value, and recent active files accessed
Secret Messages	Can't extract (No passphrase identified)	To extract the secret messages, additional consent and interview with the suspect for required passphrase is needed
Photo Upload	found	EnCase Internet Artefacts Search
Message Posted	Partly found	EnCase keyword search

4.5 CONCLUSION

This chapter reported the findings of possible steganographic techniques that can be performed on Facebook and Google+ in a laboratory environment. The research implemented two possible steganographic techniques chosen from the pre-test in two experimental case scenarios in order to study how digital forensic investigation can be conducted if image steganography is involved in online social networking. What was learnt from the experimental case scenarios is that it is important for the investigator to be aware of the various image steganographic techniques that can be used for image sharing in online social networking and their impact on the forensic examination.

The capability of automated steganographic analysis tools is very important for digital forensic examination. At present, automated steganographic detection tools are limited. The automated steganographic tool detector, StegAlyzerAS is capable of detecting steganographic tool artefacts as seen in both case scenarios, and was shown to be helpful for the investigation. However,

StegAlyserSS is incapable of detecting the steganographic content of images generated by JP Hide and Seek or StegHide. This indicates the inadequacy of steganalysis tools. JP Hide and Seek (JPHSWIN.EXE) has been established for 14 years now; yet a reliable automated steganalysis tool for its detection is still not available. Although StegDetect is able to detect JP Hide and Seek steganographic algorithms, its detection ability is still dependent on the cover image chosen. If the chosen cover image has a quantization table error, then steganographic detection will be impracticable.

Another lesson learnt from the experimental case scenarios is that artefacts found by the steganography detection tool, and from online social networks, Windows prefetch files, internet history, and HEX values significantly contribute to digital forensic analysis especially in identifying potential steganographic images. Furthermore, availability of the passphrase for secret message extraction is vital for extracting potential embedded evidence (secret message) and providing direct evidence of steganographic content. A further discussion will be carried out in the next chapter, to link the research findings to the research question, sub questions, research hypotheses and ultimately to recommend investigative steps that should be undertaken while examining image steganography associated with online social networking.

Chapter 5

Research Discussion

5.0 INTRODUCTION

Chapter 4 reported the findings of the experiments undertaken according to the research design established in Chapter 3. The findings of the experiment in Chapter 4 enabled the researcher to ascertain the methods of image steganographic exploitation in online social networking as well as the approach of digital forensic investigation in this context. Chapter 5 is now to test the hypotheses established in Section 3.2.3 and to set up a discussion that relates the research findings with the research question and the sub questions.

Section 5.1 is to gather the findings from Chapter 4 and answer the research question in which the hypotheses will be tested. Before that, the research sub questions outlined in Section 3.2.3 are to be answered and discussed in order to determine the arguments made for and against each derived hypothesis in Section 3.2.3. Each of the associated hypotheses in Section 5.1.2 and the main research hypothesis in Section 5.1.3 will be presented in table form. The justification of the hypothesis as accepted, rejected or indeterminate will be based on the arguments made in accordance to the research findings. Section 5.2 then presents a discussion of the research findings from the experiment and the expectations raised by the literature review in Chapter 2. Lastly, Section 5.3 concludes Chapter 5 as a whole.

5.1 ANSWERING THE RESEARCH QUESTION

The purpose of this section is to test the research hypotheses that were established in Section 3.2.3 against the findings that were collected in Sections 4.3 and 4.4. In order to evaluate the arguments in the research hypothesis, this section starts with Section 5.1.1 answering the research sub-questions according to the evidence collected from the experiment. Section 5.1.2 is to test the main research hypothesis and associated hypotheses with arguments for and against set out in tabular form. Arguments for support the asserted hypothesis whereas arguments

against refute the asserted hypothesis. Ultimately, in Section 5.1.3, the answer to the main research question will be found.

5.1.1 Answers to Sub-Questions

In order to answer the research question and to validate the research hypothesis, the sub-questions that derived in Section 3.2.3 need to be answered and so, the sub-questions' answers will be presented in tabular reports, Tables 5.1 to 5.9.

Table 5.1: Sub-Question 1 and Answer

<p>Sub-Question 1 (SQ1):</p> <p>Can the automated steganalysis tool StegAlyzerAS identify steganographic tool artefacts in the target's system?</p>
<p>Answer:</p> <p>Yes</p>
<p>Summary:</p> <p>Yes, StegAlyzerAA, is capable of detecting steganographic tool artefacts. The two popular steganographic tools tested in the experimental case scenarios were JP Hide and Seek and StegHide. Neither of the applications require any installation. JP Hide and Seek was used in experimental Case Scenario 1 and StegHide was used in Case Scenario 2. When JP Hide and Seek application was first downloaded from its original website, it is compressed in a zip file (Jphs_05.zip). This zip file contains all the necessary execution files for the application. In experimental Case Scenario 1, the uncompressed JP Hide and Seek application was loaded into the first target machine, whereas in the second target machine, the application was downloaded from the website directly with the application compressed in a zip file. The detection result from StegAlyzerAS showed that 80% of the file artefacts detected in the first target machine and Jphs_05.zip file artefact was not found. This result was accurate as Jphs_05.zip did not exist in the first target machine. In the second target machine 100% of the known file artefacts were detected. As for experimental Case Scenario 2, it was not surprising that StegHide application file artefacts could not be detected by StegAlyzerAS as the application was executed from a USB flash drive. However, four registry files (33.3%) associated with StegHide were identified by</p>

StegAlyzerAS. Additionally, the execution artefacts for both case scenarios were also been found in the Windows prefetch files.

Table 5.2: Sub-Question 2 and Answer

Sub-Question 2 (SQ2): Where are identified steganographic tool artefacts located?
Answer: It depends on where the user saved the console application. If the application was saved and executed from the USB flash drive it can be found in NTUSER.DAT file.
Summary: Both steganographic tools used in the experiment case scenarios are console applications. Therefore, installation is not necessary and can be executed from any directory or drive in which the user has stored the application package. The identified steganographic tool artefacts in experimental Case Scenario 1 were located in: (Target Machine 1) 1) I:\Users\Christian\Documents\Christian\jphs_05\jphs05\ (Target Machine 2) 1) R:\Users\John\Downloads\jphs_05.zip 2) R:\Users\John\Downloads\jphs_05\jphs05\ Note: Drives I:\ and drive R:\ are the mounted drives of the evidence file. The identified steganographic tool artefacts in experimental Case Scenario 2 were located in: (Target Machine 3) 1) I:\Users\machine2\NTUSER.DAT Note: Drive I:\ is the mounted drive of the evidence file.

Table 5.3: Sub-Question 3 and Answer

Sub-Question 3 (SQ3): How long does it take StegAlyzerAS to identify steganographic tools' artefacts?
Answer: It took 5 to 7 minutes to analyze a hard drive capacity of between 49.90 GB to 119.90 GB
Summary: <u>Experimental Case Scenario 1</u> (Target Machine 1) Time Begin: 2:02:29 p.m. Time End: 2:07:38 p.m. Time Elapsed: 0:5:8 Hard Drive Capacity: 49.90GB (Target Machine 2) Time Begin: 2:59:24 p.m. Time End: 3:04:42 p.m. Time Elapsed: 0:5:18 Hard Drive Capacity: 49.90GB <u>Experimental Case Scenario 2</u> (Target Machine 3) Time Begin: 7:52:49 p.m. Time End: 8:00:05 p.m. Time Elapsed: 0:7:16 Hard Drive Capacity: 119.90GB

Table 5.4: Sub-Question 4 and Answer

Sub-Question 4 (SQ4): Can StegAlyzerSS identify the uploaded and downloaded steganographic images from an OSN?
Answer: Yes for appending techniques. No for JP Hide and Seek and StegHide.
Summary: From the experimental results, StegAlyzerSS can only detect steganographic image that employ the append technique. Both JP Hide and Seek and StegHide steganographic signatures cannot be identified by StegAlyzerSS. The steganographic images in both experimental case scenarios were identified mainly with the information from other relevant artefacts such as information gathered in the interview, Facebook chat, information in the Windows prefetch files, Internet file download history, Windows Explorer file activity and multimedia activity. From the experimental experience, it was discovered that including the detection of a steganographic tool first during the survey of digital scene is important as it will prompt the investigator to look for probative information that could be hidden using steganographic techniques especially as the current automated steganalysis tools are inadequate in this regard.

Table 5.5: Sub-Question 5 and Answer

Sub-Question 5 (SQ5): Where are the identified steganographic images located in the target system?
Answer: It depends on where the user saved the generated or downloaded steganographic images.
Summary: The steganographic images in experimental Case Scenario 1 were manually identified in: (Target Machine 1) 1. D:\Users\Christian\Pictures\from John\

2. D\Users\Christian\Downloads\
3. D\Users\Christian\Pictures\Special pictures\
4. D\Users\Christian\Pictures\Photos\

(Target Machine 2)

1. D\Users\John\Pictures\To Christ\
2. D\Users\John\Pictures\from Christ\
3. D\Users\John\Downloads\special photos\

The steganographic images in experimental Case Scenario 2 were manually identified in:

(Target Machine 3)

1. D\Users\machine2\Pictures\

Please note that the location of the identified steganographic images will vary case-by-case as it depends on where the user saved or downloaded the generated steganographic images. Steganographic images will be hard to identify if the automated detection tool is inadequate.

Table 5.6: Sub-Question 6 and Answer

Sub-Question 6 (SQ6):

Is the process of determining steganographic images tell from which OSN these images were downloaded or uploaded?

Answer:

Yes. But, this is not reliable evidence as the integrity of the findings cannot be justified.

Summary:

The experimental investigation was to focus on steganographic images uploaded or downloaded from specified OSNs. Facebook artefacts can disclose which image files have been uploaded and the browser download history can disclose from which domain the same image file name is downloaded. This is applicable only if the user does not change the image file name after uploading the image or when saving the downloaded image. This is significant as the MD5 value of the steganographic image identified cannot be verified against the Facebook artefacts found as the artefacts only indicate the file name, not the actual file. In Facebook there are two associated download domain; one can be found as

http://www.Facebook.com/download/[file unique id numbers]/ [filename] and the other one as http://attachment.fbsbx.com/file_download.php?id=[file unique id numbers]&eid=[encrypted link name]&ext=[unix time]&hash=[unique hash value].

Google+ is different from Facebook because Google+ photo sharing enables image steganography, so if the identified steganographic image was involved in suspect Google+ content, these images are cached in the browser cache file. Therefore, one is able to tell the identified images are from Google+. However, the integrity of the file cannot be verified because the MD5 value of the images cached in the cache file does not match the identified steganographic images. It only indicates the same image was posted on Google+ user content.

Table 5.7: Sub-Question 7 and Answer

Sub-Question 7 (SQ7):
How long does StegAlyzerSS take to identify steganographic images?
Answer:
It took less than one minute to establish in the where suspect drive the steganographic images were located with hard drive capacities ranging from 49.90 GB to 119.90 GB.
Summary:
<u>Experimental Case Scenario 1</u>
(Target Machine 1)
Scan Finished in: 0:0:43
Hard Drive Capacity: 49.90GB
(Target Machine 2)
Scan Finished in: 0:0:47
Hard Drive Capacity: 49.90GB
<u>Experimental Case Scenario 2</u>
(Target Machine 3)
Scan Finished in: 0:0:58
Hard Drive Capacity: 119.90GB

Table 5.8: Sub-Question 8 and Answer

Sub-Question 8 (SQ8): Can StegAlyzerSS extract the secret message embedded in the images?
Answer: No. Not all embedded secret messages in the steganographic images identified can be extracted by StegAlyzerSS.
Summary: Based on the experiment, steganographic techniques that appended secret messages in an image can be extracted whereas steganographic techniques in JP Hide and Seek and StegHide cannot be extracted by StegalyzerSS because neither could be detected by StegAlyzerSS.

Table 5.9: Sub-Question 9 and Answer

Sub-Question 9 (SQ9): How long does it take StegAlyzerSS to extract the secret message?
Answer: The time taken for secret message extraction cannot be determined.
Summary: The time taken to extract the secret message is undetermined at this stage as StegAlyserSS was unable to identify the JPHide and StegHide steganographic signatures in the images and were therefore, unable to extract the embedded secret message. Although, the appended technique in experimental Case Scenario 1 was able to be detected by StegAlyzerSS and its hexadecimal analysis was able to search for the appended data at the end of file (Figure 4.14), the exact time taken to extract the secret message is still undetermined as the process of extraction is not automated. However, manual extraction conducted in the experiment took less than a minute to search for the end of file signature.

5.1.2 Hypotheses Testing

There are three associated hypotheses to be tested in order to verify the validity of the research findings and to answer the research's main question. These hypotheses will be tested with arguments made for and against to either prove or refute the tested hypotheses with supporting evidence obtained from the

experimental case scenarios. The tested hypotheses are presented in Tables 5.10 to 5.12.

Table 5.10: Tested Hypothesis 1

Hypothesis 1 (H1): When conducting a digital forensic examination, the footprint of a steganographic tool or its usage can be identified.	
TEST RESULT: Indeterminate	
ARGUMENT FOR: <p>The steganographic tools, JP Hide and Seek and StegHide used in both the experimental case scenarios were able to be detected by StegAlyzerAS. An evaluation of steganography was included in an early stage of the digital forensic investigation procedure called the survey of the digital scene. StegAlyzerAS was able to indicate the location of the steganographic application stored. Additionally, the time that StegAlyzerAS took to evaluate and detect the steganographic tool in the target systems was relatively short, ranging from 5.08 minutes to 7.16 minutes. This however was dependent on the target hard drive data size.</p> <p>The execution of the steganography tool could also be traced from the Windows prefetch files in the target system. Windows prefetch file evaluation can verify the usage of the</p>	ARGUMENT AGAINST: <p>Although steganographic evaluation is included in digital forensic investigation procedures, it can not guarantee that all steganographic tools can be detected by automated detection tool. If the detection tools is not updated with the latest steganographic techniques or it does not included the signatures of such steganographic tool, detection may not be successful. Furthermore, there may be steganographic algorithms that are unknown or publicly unavailable. Steganographic tool detection is similar to antivirus applications if the application is not updated with carras viruses, the antivirus application will not be able to detect the latest viruses.</p> <p>Moreover, when the steganographic tool is portable, the amount of significant registry artefacts detected is not convincing as of 12 registry artefacts only four were detected</p>

<p>steganographic tool once the steganographic tool was detected at the early evaluation stage of the investigation. The findings from both experimental case scenarios did indicate the execution of the steganography tool used (Figures 4.10, 4.11 and 4.18). Windows prefetch files not only indicated the execution of the application installed in the target's system but also captured the execution of applications running from a portable device like USB flash drives. Looking at the Windows prefetch files may as well provide valuable information to the investigator especially when the steganographic detection tool is unable to detect the existence of steganographic tools in the initial evaluation.</p>	<p>(Appendix 29)</p> <p>Although the experiment showed that StegHide activities were captured in the Windows prefetch file, this information was limited as it could be overwritten over time. It only included up to 128 prefetch files in the prefetch folder. When the entries exceed 128 entries, Windows will automatically overwrite the prefetch file entries in the folder (Sutherland, Evans, Tryfonas, & Blyth, 2008). Therefore, searching for such information in prefetch files may not be successful when an investigation is conducted same time after the event.</p>
<p>SUMMARY:</p> <p>The steganographic tools implemented in both the experimental case scenarios were successfully detected by StegAlyzerAS and the footprints of steganographic tool execution were captured in the Windows prefetch folder as well. However, the automated tool has some limitations being the tools' signatures were undefined or unknown. Although Windows prefetch files provided valuable information in regard to the research experiment, Windows prefetch folders also have their limitations. Therefore, the arguments made for and against prove the hypothesis indeterminate.</p>	

Table 5.11: Tested Hypothesis 2

<p>Hypothesis 2 (H2):</p> <p>When conducting a digital forensic examination, the steganographic images can be identified.</p>	
<p>TEST RESULT:</p> <p>Accepted</p>	
<p>ARGUMENT FOR:</p> <p>The automated StegAlyzerSS was able to identify steganographic images that had appended the secret message at the end of file. Although JP Hide and Seek and StegHide steganographic signatures in the experimental images could not be identified by StegAlyzerSS, the surrounding relevant information gathered during the digital forensic examination and analysis were able to lead the investigator to identify the steganographic image manually. For example, in Case Scenario 1, Facebook chat (Table 4.8), social network upload (Figure 4.13), download artefacts (Appendices 15 & 16), windows explorer multimedia and file activities (Figure 4.17) were significant in leading the investigator to identify the suspect steganographic images. Moreover, the steganographic tool identified in the early stage of the investigation also gave a hint of the steganographic algorithm used by the suspect, therefore enabling the investigator to look for an appropriate</p>	<p>ARGUMENT AGAINST:</p> <p>The identification of steganographic images may be limited when surrounding relevant evidence is unavailable to the investigator.</p> <p>In experimental Case Scenario 2, even though the image captured in the browser cache file and the suspect steganographic image seemed to be the same image (Figure 4.22), both images actually have different bit streams as the hash MD5 values of the images are different. The image captured in the browser cache file has been changed by the browser when it downloaded the image from the social network server. Hence, there is no evidence to prove the content of both image files are exactly the same.</p> <p>The content of the steganographic images is vital as the probative evidence is embedded in the image and cannot be detected directly by the human eye. Therefore, if the embedded evidence has been manipulated by the</p>

<p>steganalysis tool, StegDetect, to confirm the existence of embedded digital evidence (Figures 4.15, 4.16). Similarly, in the experimental Case Scenario 2, surrounding relevant evidence lead the investigator to identify the steganographic images. For example, the image file and text file that were traced in the StegHide prefetch file (Figure 4.19), the unusual HEX values that appeared in the suspect image files (Figure 4.20), images in the browser cache file that were the same image as the suspect steganographic images (Figure 4.22), the activities on the suspect steganographic image file accessed after the user accessed the confidential data (Figure 4.23) and lastly the timeframe of image upload activities coinciding with the period of steganographic activity (Figure 4.24).</p> <p>What was discovered in the experimental case scenarios was that identification of suspicious steganographic images can be done manually by analyzing indirect evidence rather than depending on the results of automated tools, which can be rather inadequate.</p>	<p>browser application and stored in the cache file, this evidence is inadmissible in a court of law. The evidence cannot prove that a specific image in the social network content is the steganographic image identified on the suspect's machine as the MD5 hash values are not the same. It can only prove that the same image is found in the user's social network content.</p>
--	---

SUMMARY:

In the experimental case scenarios, although the identification of steganographic images was dependent on available information gathered in the digital environment, based on the experimental scenarios and investigation procedures conducted in the research, steganographic images were in fact being identified. Although there is an argument made against the integrity of the image identified in browser cache file. Other than that, the steganographic tool identified, the steganographic tools' artifacts found in the Windows prefetch folder, the unusual HEX value of the suspect steganographic images, the suspicious multimedia and file activities in Windows explorer, the Internet upload and download activity, and the timeline analysis identified manually the steganographic images. Therefore, the arguments made for and against confirm that the hypothesis is to be accepted.

Table 5.12: Tested Hypothesis 3

Hypothesis 3 (H3): The hidden data in identified steganographic images can be extracted when conducting a digital forensic examination.	
TEST RESULT: Indeterminate	
ARGUMENT FOR: In Case Scenario 1, the data embedded in the steganographic images identified were successfully extracted because both the passphrase and steganographic algorithm used by the suspects had been identified. When a passphrase is not available to extract the secret message, the investigator can search for file types that can be embedded using the	ARGUMENT AGAINST: Although the embedded evidence could be extracted in Case Scenario 1, it heavily depended on the passphrase being identified by the investigator. If no hint can be found, the investigator would have a hard time extracting the embedded data. This situation was relevant to Case Scenario 2 when the embedded data could not be revealed due to there being

steganographic algorithm the suspect's machine and look for suspicious file content that could have been extracted by the suspect himself or herself (Figure 4.17).	no passphrase hint. Further cryptanalysis would be needed to be performed when the embedded secret message was unable to be extracted during the forensic examination.
SUMMARY: The extraction of embedded data in the steganographic image is proven to be dependent on the available information. Insufficient information will make it hard for an investigator to extract the embedded secret message during a digital forensic examination. The extraction of embedded data is vital when the secret message is the direct criminal evidence. The arguments made for and against show that the hypothesis is indeterminate.	

5.1.3 The Research Question Answer

Table 5.13 is the main research question and the main hypothesis that was to be tested based on the answers gathered from the research sub-questions and the associated hypotheses tested in Section 5.1.1 and Section 5.1.2 respectively.

Table 5.13: Research Main Question and Tested Hypothesis

Main Question: <i>Should digital forensic investigators include steganography as part of their routine check in the standard procedure of digital forensic investigation in relation to online social networks?</i>	
Main Hypothesis: <i>That digital forensics investigator should include steganographic evaluation as a routine check in their standard digital forensic investigative procedures in relation to online social networks as the footprints of steganographic tool, its usage, and the steganographic image can be identified.</i>	
TEST RESULT: Accepted	
ARGUMENT FOR: Both experimental case scenarios conducted in the research included	ARGUMENT AGAINST: Although the steganographic activities in both experimental case scenarios

<p>steganographic evaluation in the digital forensic investigation procedure. Although they were different experimental case scenarios, based on the outlined investigative procedures, the steganographic images related to the online social networks were able to be identified.</p> <p>First, the steganographic tool was identified. Then, the steganographic images were identified with the applicable automated steganalysis tools or manually identified with an appropriate technique (Sections 4.3 & 4.4). Moreover available relevant evidence such as online social network artefacts, Internet upload or download artefacts, multimedia and file activities gathered from the target machines and timeline analysis were shown to be related to the suspected steganographic objects.</p> <p>The extraction of embedded evidence can be done dependent upon the availability of information. In scenario 1 embedded data could be extracted whereas in Scenario 2, embedded data could be extracted as there was no hint for the investigator about the passphrase. However, further extraction is possible with additional consent to</p>	<p>were able to be identified, manual identification of steganography is time consuming. If an automated steganalysis tool could be loaded with all available steganographic's signatures and were able to analyze and detect steganographic images then it would speed up the investigation and it would be worthwhile including it as a routine check.</p> <p>While steganographic images can be identified, the ability to extract the embedded incriminating data is crucial in digital forensics. Investigators still face the possibility of failing to extract the direct evidence, therefore the evidence may still not be sufficient to prove the crime.</p>
--	---

<p>get the passphrase or by performing cryptanalysis. Cryptanalysis is thought to be possible because both of the experimental case scenarios have known steganographic algorithms, known secret messages, and known stego-objects in the suspects' machines.</p>	
<p>SUMMARY:</p> <p>The steganography activities implemented in the two experimental case scenarios were successfully identified because steganographic evaluation was included in the digital forensic investigation procedure. Otherwise, steganographic images residing in the suspect's system will go seem to be ordinary online social network activities such as image uploading and downloading. There may be nothing interesting to lead the investigator to think that the case involved steganography unless there are hints from social network artefacts to indicate such an involvement, for example messages posted or live chat such as Facebook chat. Although there are limitations; it being time consuming, manual work, unavailable information, unknown steganographic algorithms; the experiment conducted in the studies positively showed that steganography can be undertaken in online social networking and it leaves behind footprints which can be identified. Moreover, embedded data, which can be the valuable direct evidence, is possible to decrypt. Therefore, the arguments made for and against suggest the main hypothesis is to be accepted.</p>	

5.2 DISCUSSION

This section is to focus on the significant findings that have been discovered in the digital forensic investigation procedures deployed in Phase 3 and Phase 4 of the research and how the different digital environments set up in Phase 2 affected the steganographic investigations. Section 5.2.1 is to discuss how the case scenario environment affected the steganographic investigations. Sections 5.2.2 and 5.2.3 are to discuss the difficulties that occurred during the steganographic evaluation in the digital forensic investigation with reference to the literature review studied in Section 2.5. Lastly, Section 5.2.4 is to recommend the procedure for

steganographic evaluation that can be used in similar environments. Research Phase 1 will not be covered in this section as Phase 1 was to test and understand the types of steganographic techniques currently supported by the two most prominent online social network platforms.

5.2.1 Discussion of the Case Scenario Environment

The experimental case scenarios were set up to be as close as possible to the real world Windows environment. A difficulty was encountered when setting up the experimental Case Scenario 1, where the scenario was intended to contain Facebook chat artefacts that would assist the investigation. However, this artefact could not be generated after a number of attempts. The literature reviewed in Chapter 2 indicated that Facebook chat could be discovered in a browser cache file. However, after two simulation attempts, Facebook chat 1 could still not be discovered in a browser cache file. Finally, it was discovered that Facebook chat was no longer cached in the browser cache file and could only be found in pagefile.sys, which is the virtual memory file, hibernation file or the unallocated cluster. So, from the experiment environment set up, it was discovered that not all live chat artefacts can be found in a suspect's hard drive, as the information contained in the pagefile.sys, hibernation file or unallocated cluster is generated over time. Thus, sometimes live memory forensics may help to obtain current online social networking artefacts that may not be found in the target's hard drive. However, the consequence of live memory forensics is that the artefacts from the acquisition process can be left in the target's system and thus does not preserve the integrity of the target's system (Savoldi, Gubian, & Echizen, 2010). Furthermore, live memory forensics is literally only applicable when the system is live.

The scenarios in the research experiments were set up in the Windows 7 environment and the online social network activities were performed using Internet Explorer version 8. In the real world, the steganographic algorithm, operating system and internet browser encountered by an investigator may be different, therefore where the artefacts reside in the system may also be different. The simulated research environments were able to stress the possibility of steganography in online social networking and were sufficient to highlight the importance of steganographic evaluation in the digital forensic investigation

procedure in general. Both the experimental case scenarios proved that, in general, when steganography is involved, the target system would likely contain the steganographic algorithm and steganographic object and sometimes may even have the cover-object available, as in Case Scenario 1.

5.2.2 Discussion on Data Acquisition and Extraction

The data acquisition and extraction conducted in Phase 3 of the research used appropriate tools to acquire and extract the relevant evidence according to the evaluation and assessment of each case. The experimental case scenarios were intended to evaluate investigative procedures for steganography involved in online social networking. Therefore, the data was acquired and extracted with regard to social network forensics and web browser forensics and best practices were applied as from the literature review in Sections 2.5.1 and 2.5.2.

In Section 2.5.1, Mulazzani, Huber, and Weippl (2012) mentioned that social footprints such as the user's social network friend connections, communication patterns, what has been uploaded and who has been tagged, and the time of activity are importance sources of viable evidence, which is proven based on the experiment conducted. However, the authors stated that the information cannot be found in the hard drive and the information is only stored at social network provider's site. From the experiment conducted it was found that, although the information is not specifically stored in the suspect's hard drive, similar information can still be extracted from pagefile.sys, unallocated clusters, java script files and so on (Appendix 19, 20, 22, 23) that is stored in the hard drive. And from the information extracted, the investigator is able to identify the user's social network friend connections, their communication patterns, what has been downloaded, and the execution time of such activity, which was very helpful for the investigator to further reconstruct and analyze the connection between the information extracted and its relevance to steganographic involvement in online social networking.

Web browser forensics is also another important process to extract relevant online social network activities and to identify steganographic communication. It was reported in the literature review that web-based chat could be found in browser cache files (Mutawa et al., 2011). However, from the experiment conducted, instant web-based chats (Facebook chat) were extracted

from pagefile.sys and unallocated clusters and none were found in the browser cache file (Appendix 22). The availability of this information is volatile as the information contain in the pagefile.sys is actually dependent upon the Windows system configuration. If the pagefile.sys is configured to be ‘turned off’, then valuable information in the pagefile.sys may not be found. Other than that, the evidence that was extracted with web browser forensics significantly contributed to identifying the online social network activities as well as identifying steganographic communication (Appendix 15, 16, 17, 18, 24, and 25).

5.2.3 Discussion on Reconstruction & Analysis

Based on the findings, it was proven that the information available as reported in Section 5.2.2 significantly aided the investigator to identify the steganographic information posted using online social networking. The information not only aided the investigator to map the occurrences in the case but enabled evidence collection. The information contains important meta data that is captured for the court report. For example, the unique message ID, the sender name and profile number, the recipient name and profile number, and the date and time in regard to the message can be clearly extracted. This information was proven in Case Scenario 1 where these meta data could be gathered from Facebook artefacts left in the suspect’s system. However this data was lacking in Google+, which may be due to the different environment set up in Case Scenario 2.

When comparing Case Scenario 1 and Case Scenario 2, valuable artefacts from social network content in Case Scenario 1 could be obtained from the pagefile.sys, but nothing could be found in Case Scenario 2. The pagefile.sys is a virtual memory swap file where data are swapped out of RAM and stored in this file during the system’s normal operation. This file is volatile and the information stored in the pagefile.sys is generated over time and cannot be gained in a quick process as in Case Scenario 2. This may be part of the reason why pagefile.sys in Case Scenario 2 did not capture any social network content from Google+.

As mentioned in Section 2.5.3 the goal of steganalysis is to “identify suspected information streams, determine whether or not they have hidden messages encoded into them, and if possible, recover the hidden information” (Kumar & Pooja, 2010, p.21). In the experiment, although the steganographic images could be identified and the steganographic algorithm could be determined,

the process of identifying the steganographic objects was very challenging as the automated steganalysis tool initially selected – StegAlyzerSS, was found to be inadequate in identifying the steganographic technique used in the experiment. Even though the cover-object and stego-object were available on the target's machine; visual detection was impracticable because it is hard for the human eye to perceive such difference. Furthermore, data size of an image cannot show which is a steganographic image. As the example showed in Section 2.3.5, steganographic images can have smaller file sizes than the cover-image or vice-versa. Therefore, it was discovered in the experiment, identification of the steganographic tool at an early stage in the investigation aids the steganalysis process. On the other hand, the evidentiary trails discussed in Section 5.2.2 significantly aided the reconstruction and analysis of the digital forensics investigation. Additionally, HEX value analysis is another feasible technique for identifying unusual images patterns manipulated using steganography (Figure 4.20 and Figure 4.21).

It was understood from the literature review in Chapter 2 that steganalysis includes the hidden message destruction (Ibrahim, 2007); and yet from the experiment it is found that this statement cannot be applied. When steganography does happen, from the digital forensic point of view, recovery of the hidden message is significantly more important than destroying the embedded data as it is possible incriminating evidence. Destruction is not feasible during a digital forensic examination as the rule of thumb in digital forensics is to preserve the integrity of digital evidence so that the identified evidence is admissible in a court of law. This was true in both the experimental case scenarios where in Case Scenario 1, the embedded secret message was a terrorism related action plan whereas in Case Scenario 2 it was corporate espionage information. Therefore, from a digital forensics perspective, the detection and recovery of embedded secret messages are to be equally important in the reconstruction and analysis phases. When the embedded information is not able to be extracted, other relevant footprints giving evidence that the identified objects have been steganographed must be given. Otherwise, the suspected steganographic object can be analysed for reverse engineering or decrypted by experts for possible extraction. As quoted in Section 2.5.3, according to Ibrahim (2007, para.13) there are four situations where secret message recovery is possible: “1) when only the steganographic

object is available, 2) when the steganographic algorithm is known and the steganographic object are available, 3) when the steganographic object and the original cover object is available, 4) when both the steganographic and the cover object are available and the steganographic algorithm is known.” Two of these situations were identified in the findings. In Case Scenario 1, both the steganographic and the cover object were available and the steganographic algorithm was known, whereas in Case Scenario 2, the steganographic algorithm was known and the steganographic object was available.

5.2.4 Recommendation for Steganography Evaluation

The flow chart diagram in Figure 1.1 is a reflection of the experimental case scenarios and the literature review in Chapter 2. The flow chart has proven practicable for steganographic evaluation associated with online social networking and the investigation was conducted in a forensically sound manner with the integrity of the digital evidence preserved (Appendix 3, 4, 5, 6, & 7). This research was to emphasize that every digital forensic investigation procedure should include steganographic evaluation. Michael Sheetz, in his article recommended that investigator approach every investigation with the assumption that steganography can benefit the suspect (Sheetz, 2003). This approach has been tested in both the experimental case scenarios (Section 4.3 and section 4.4) and proven to be accepted (Table 5.12).

Additionally, in the Handbook of Information Security, the author of the chapter, *Computer Forensics Procedures and Methods* emphasized signature analysis to identify files that are hidden from plain sight by changing their file extensions (Craig, 2006). This evaluation is included in standard computer forensic procedures and has been proven to help identify obscured file types. However, criminal technology can improve over time and criminals can learn from their mistakes. Therefore, criminals may look for more advanced techniques to hide from plain sight. Utilizing steganographic tool to hide from plain sight does not require advanced technological knowledge. Anyone capable of changing the file extension is capable of using a steganographic tool. Thus, the research findings suggest the investigator add additional steganographic signature analysis to the standard computer forensic procedures. The only significant challenge

discovered in the recommended steganographic evaluation is the steganalysis tools selected. These tools could identify some of the steganographic signatures found in the experimental case scenario. If this limitation can be overcome, the recommended steganographic evaluation can be more effective because automated procedures can reduce manual evaluation time. The recommendations for best digital forensic investigator practice derived from the research and the literature are summarised in Figure 1.1.

5.3 CONCLUSION

This chapter has discussed the research findings according to the research experiments reported in chapter 4. The research question and sub-questions derived in Chapter 3 have also been answered based on the findings as shown in Sections 4.3 and 4.4. The asserted hypotheses were then tested accordingly with arguments made for and against in order to see whether the asserted hypothesis is to be accepted, rejected or indeterminate. The difficulties and limitations encountered in the research experiment have also been discussed.

The main objective of the research was to determine whether or not steganographic evaluation should be included in the standard digital forensic procedure. Possible steganographic techniques used on online social networks have been experienced and the impact on the digital forensic investigator has been recorded. The research experiment and observation have positively shown the importance of including steganographic evaluation in the standard procedure. Lastly, the steganographic evaluation performed in the experimental case scenarios was presented in an easy to understand flow chart diagram as a guideline for future reference. The next chapter, Chapter 6, will conclude the thesis by presenting the significant research findings (see Figure 1.1). Potential future research will also be outlined so that others may further develop what has been reported here.

Chapter 6

Conclusion

6.0 INTRODUCTION

This chapter is to summarize the entire thesis project and to draw a final conclusion based on the research findings of Chapter 4 and the discussion in Chapter 5. Research difficulties and limitations that were encountered are reported and the gaps identified in the current research are reported as opportunities for further research.

Image steganographic technique is the main consideration of this research project. While steganography is a form of security through obscurity in the security world, it is also a form of security whomever wishes to perform nefarious deeds where they are able to conceal incriminating evidence or to perform covert communication without being identified by law enforcement or the relevant authority. This threat has raised the awareness of digital forensic investigators in this context as steganography cannot be found if it is not looked for. The emergence of social networks has created a new means of communication. Therefore, when image steganography integrates with this new communication platform, a new threat to digital forensics has evolved, as online social networking has increasingly been used as a tool to perpetrate crime. Due to the lack of investigative guidelines and procedures in this context, the objective of the research was not only to discover appropriate evaluation procedures in this context, but also to measure whether steganographic evaluation procedures should be included in the standard procedures of digital forensic investigation. A research methodology was designed to fulfill the research objective and to ensure that the research was conducted using a reliable method based on previous relevant studies.

In Chapter 4 the results of the research phases were reported. In Phase 1 of the research, five steganographic techniques using common image formats, JPEG, GIF, BMP and PNG were tested on the two most popular social network platforms, Facebook and Google+. This phase was to assess the features of the

selected social network websites that supported or inhibited the propagation of steganographic images. Therefore the preliminary findings from Phase 1 enabled the researcher to identify and understand the techniques that exploit image steganography in online social networks.

Phase 2 of the research was to apply the most common and easy to get steganographic technique, JP Hide and Seek and StegHide discovered in Phase 1, to two different experimental case scenarios respectively. The commonly used JPEG image format was selected as the carrier format for steganographic manipulation. The objective of the simulations was to discover the footprints left behind after the simulation of each case scenario. Once the simulations were complete, Phases 3 and 4 were executed. Phase 3 and phase 4 were not only designed to acquire, extract, and analyze digital evidence left behind by Phase 2, but also to discover an effective digital forensic procedure for evaluating image steganography associated with online social networking. Therefore, digital forensic processes in the experiment were carried out in a forensically sound manner to ensure the validity of the collected evidence.

Subsequent to the processes conducted, the experimental findings proved that a steganographic tool, steganographic images, and secret message was successfully discovered in Case Scenario 1. Additionally, there were also other significant artefacts could back up the findings of Case Scenario 1 such as the steganographic tool artefacts identified in the Windows prefetch files, social network artefacts such as image files, internet browser history that indicated image download activities and other text files and multimedia files that were captured in the Windows Explorer history.

In Case Scenario 2, the steganography tool's registry artefacts were identified by StegAlyzerAS, but not many registry artefacts were recovered because StegHide was executed from the USB flash drive. However, by using cross checking techniques, the artefacts were also found in the Windows prefetch folder, thereby, showing that the steganographic tool was used on the suspect system. The next most difficult part of Scenario 2 was the extraction of the embedded data. This action was not able to be performed due to insufficient information about the passphrase. The option of password cracking may be taken or additional information from further interviews may help recover the hidden message. Although the secret message could be recovered, there were other

footprints which gave evidence that the identified images may have been manipulated by steganography, such as the abnormal HEX value of the identified images, the image files contained in the StegHide prefetch file, the sequential pattern of files accessed in Windows Explorer history reflected the steps of the image steganography embedding process, lastly, in the browser cache file, the same file names of the suspected images were discovered in the social network URL and also displayed the same image.

The findings from Phase 3 and Phase 4 have helped to answer the research sub-questions, provide evidence for the tested hypotheses and thereby answered the main research question. Based on all evaluation and findings from the research experiment, the ultimate answer to the research question is that steganography evaluation should be included in routine checks and standard procedures for digital forensic investigation in relation to online social networks. However, steganalysis tools assessed in the evaluation require further improvement to include the latest steganographic signatures so that all steganographic images can be identified and the automation of processes can be possible. Ultimately, the steganographic evaluation process of the case scenarios is recommended in Phase 5 and presented in a flow chart diagram so that it is easy to follow. This recommended process is reliable as it has been proven capable of identifying the steganographic techniques and steganographic images that are relevant to online social networking. However, the extraction of secret messages is still dependent upon available information.

6.1 LIMITATION OF RESEARCH

Some research limitations were predicted in Section 3.4 during the forming of a research methodology based on the proposed research design and data requirements in Chapter 3. These predicted limitations are discussed in this section. In addition, the limitations that were found in the research findings in Chapter 4 and the ones discussed in Chapter 5 will be summarized and discussed in this section.

The limitations in Section 3.4 indicate that the experiment is limited to image steganography only and the selected steganographic techniques whereas there are many more steganographic techniques that have not been tested. The

steganographic techniques used in the chosen social network may affect the findings of the digital investigation as not all steganography tool signatures or steganographic signatures can be identified by the chosen detection tool. This was noted in the experimental findings, where the steganographic signature of both JPHide and Seek and StegHide could not be identified by the selected steganalysis tool. The US National Institute of Justice also mentioned in their website that “newer steganography-encoding techniques are being rapidly developed rendering the current detection tools ineffective” (National Institute of Justice [NIJ], 2010, para.7)

Similarly, the investigation was performed on the two most popular social networking websites whereas there are more than 100 major active social networking websites which are currently available globally, excluding dating social networking websites (“List of social networking websites,” 2013). Each social networking website may have different architecture to organize their user generated content and which may affect the artefacts that could be left behind. This was apparent in the research findings from experimental Case Scenario 1 that was performed in Facebook, and Scenario 2 that was performed in Google+. When examined, Facebook artefacts and user-generated content artefacts such as image files uploaded to Facebook were identified (Appendix 19 & Appendix 20), whereas this type of information could not be identified in Case Scenario 2 – the Google+ platform. There are two possibilities why such artefacts could not be identified in Case Scenario 2; first, the social networking websites render their user-generated content differently; secondly, the duration of the simulation process of both case scenarios may have affected the content-generated artefacts. When such artefacts (Appendix 19 & Appendix 20) are important to the investigation this may affect the investigations outcome.

The other limitation that could have affected the research finding was the cover image used for the steganographic process. StegDetect is normally used as the detection tool for the JPHide and Seek algorithm where its detection capability has been proven. However, a limitation was discovered while undertaking the experiment. It was found that if the steganographic process used a cover image that has been resized, rotated, or cropped, StegDetect will have difficulty analyzing the image due to the undefined quantization table (Appendix 1).

Therefore, the tool will be incapable of detecting the steganographic signature as intended. This limitation could significantly affect the investigation findings.

The availability of relevant footprints was also shown to be a limitation as this information was vital to the research outcome. Footprints can be used as backup to prove steganography when the detection tool is incapable of identifying the signature and when extracting the secret message is not possible. This limitation was critical in experimental Case Scenario 2 when the other available sources of relevant information such as files captured in StegHide's Windows prefetch files, StegHide tool artefacts, co-related time analysis, the unusual HEX value pattern and so on were very important evidence for the case. When such information is not available, there will not be sufficient evidence to prove the suspected image files contained hidden data.

The research findings have been limited to the results from one steganalysis tool - StegAnalyzer, where other commercial steganalysis tools such as StegoSuite from WetStone Technologies recommended by the National Institute of Justice have not been used (NIJ, 2010). This may have affected the scope of the reported findings also. Besides this, the secret message embedded in the suspected steganographic images in experimental Case Scenario 2 were not sent for reverse engineering or cryptanalysis as this was not included in the scope of the research and such processes require a specialized knowledge in the area.

Lastly, the research findings were limited to the Windows 7 platform only. This may have affected the research findings when similar experimental case scenarios are simulated on different operating systems or on Windows operating systems earlier than Windows XP, as these operating systems may not have the Windows prefetch files that significantly contributed to the investigation in the research. Different operating systems may have different tools and techniques for extracting the digital evidence as the file structure of each system may vary. This variation is likely to occur in the various types of browser also.

6.2 FUTURE RESEARCH

In this research project, six steganographic techniques have been tested on two popular social networking platforms, Facebook and Google+. For future research other steganographic techniques such as text steganography or video

steganography can be tested on social networking platforms to identify the ability of social networking to support such steganographic activity. Future research could also focus on how various image steganographic techniques perform on a single social networking platform or how a single steganographic technique performs on various social networking websites to discover the digital forensics tools and techniques for preserving, collecting, extracting and analysing each different situation.

The second significant area for future research is steganalysis tools. NIJ highlighted that the development of steganalysis tools is always behind the development of steganographic tools, making steganalysis tools inadequate, especially in identifying steganographic objects (NIJ, 2010). The detection and evaluation of steganography will be more effective if an effective steganalysis tool can be developed. A steganalysis tool needed to be updated frequently and developed on par with steganographic tools so that it can detect a wide variety of steganographic signatures. If there is an automated tool that can be used to analyze all common signatures known, this could help streamline the steganographic evaluation process.

The recovery of the secret message is another significant area for future research because, when dealing with crime, especially in digital forensics, very likely the embedded data is critical evidence. Therefore, reverse engineering or cryptanalysis in the area of steganography is encouraged for future research. This is possible because frequently, steganographic algorithms, steganographic objects, and even cover objects can be found on the suspect's machine. So, assuming a password cracking tool has the option of having this available information input, can the tool be further developed into a steganography decryption tool using a common method of attack such as dictionary attack or brute force to obtain the password? This is the area where the further research could be undertaken and could possibly develop an automated steganography secret message extraction tool.

Furthermore, a further exploration can be set up in a live environment and a live memory dump can be done for memory forensics. Memory forensics may sometimes produce valuable information that cannot be found on a system's hard drive. Therefore, a live environment can be tested to figure out whether the valuable information such as the passphrase for the steganographic encryption and

other evidentiary trails which could not be found in the research findings could be gathered through the live memory to aid the investigation. This examination may well help to improve the steganography evaluation procedures as shown in Figure 1.1.

Future research can also consider forensic readiness for steganography in the corporate world. Is the corporate network prepared for such a situation? Can further information be extracted from the company's network as additional evidence to the investigation?

Although steganography is a complex investigation, it is an area that needs to be further researched and prepared for. It is an anti-forensic technique that can be used by the criminal to cover their tracks. Therefore, the researcher encourages further research in the area of steganalysis so that law enforcement and the digital forensic teams are prepared and have appropriate tools and techniques to perform such an investigation.

REFERENCES

- Acohido, B. (2011). Sex predators target children using social media. *USA Today*. Retrieved April 25, 2012, from http://www.usatoday.com/tech/news/2011-02-28-online-pedophiles_N.htm
- Alazab, M., Venkatraman, S., & Watters, P. (2009). Effective digital forensic analysis of the NTFS disk image. *UbiCC Journal*, 4(3), 551–558. Retrieved from http://www.ubicc.org/files/pdf/3_371.pdf
- Alharbi, S., Weber-Jahnke, J., & Traore, I. (2011). The Proactive and Reactive Digital Forensics Investigation Process: A Systematic Literature Review. *International Journal of Security and Its Applications*, 5(4), 59–72. Retrieved from <http://www.earticle.net/article.aspx?sn=158919>
- American Psychological Association. (2012). *APA databases methodology field values*. Retrieved October 11, 2012, from <http://www.apa.org/pubs/databases/training/method-values.aspx>
- Ashok, J., Raju, Y., Munishankaraiah, S., & Srinivas, K. (2010). Steganography: An overview. *International Journal of Engineering Science and Technology*, 2(10), 5985–5992. Retrieved from <http://www.ijest.info/docs/IJEST10-02-10-100.pdf>
- Bandyopadhyay, S. K., Bhattacharyya, D., Ganguly, D., Mukherjee, S., & Das, P. (2008). A tutorial review on steganography. *International Conference on Contemporary Computing*. Retrieved from http://www.jiit.ac.in/jiit/ic3/IC3_2008/IC3-2008/APP2_21.pdf
- Berg, G., Davidson, I., Duan, M., & Paul, G. (2003). Searching for hidden messages: Automatic detection of steganography. *Proceedings of IAAI 2003*, 51–56. Retrieved from <http://www.aaai.org/Papers/IAAI/2003/IAAI03-007.pdf>

- Berghel, H., Hoelzer, D., & Sthultz, M. (2006). Data hiding tactics for Windows and Unix file systems. *Identity Theft and Financial Fraud Research and Operation Center*. Retrieved July 22, 2012, from http://www.berghel.net/publications/data_hiding/data_hiding.php
- Carrier, B. D. (2003). Defining digital forensic examination and analysis tools using abstraction layers. *International Journal of Digital Evidence*, 1(2), 1–12. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.14.9813&rep=rep1&type=pdf>
- Carrier, B. D. (2009). Digital forensics works. *IEEE Security & Privacy Magazine*, 7(2), 26–29. doi:10.1109/MSP.2009.35
- Carvey, H. A. (2012). *Windows forensic analysis toolkit*. Massachusetts, USA: Syngress Publications.
- Casey, E. (2004). *Digital Evidence and Computer Crime* (2nd Ed.). London, UK: Academic Press.
- Castiglione, A., Cattaneo, G., & De Santis, A. (2011). A forensic analysis of images on online social networks. *2011 Third International Conference on Intelligent Networking and Collaborative Systems*, 679–684. doi:10.1109/INCoS.2011.17
- Castiglione, A., D'Alessio, B., & De Santis, A. (2011). Steganography and secure communication on online social networks and online photo sharing. *2011 International Conference on Broadband and Wireless Computing, Communication and Applications*, 363–368. doi:10.1109/BWCCA.2011.60
- Chapman, G. (2011, May 13). Social networks hotbeds for cybercrime, says Microsoft. *The New Zealand Herald*. Retrieved from www.nzherald.co.nz/technology/news/article.cfm?c_id=5&objected=10725364

- Cheddad, A., Condell, J., Curran, K., & Mc Kevitt, P. (2010). Digital image steganography: Survey and analysis of current methods. *Signal Processing*, 90(3), 727–752. doi:10.1016/j.sigpro.2009.08.010
- Chorein, A. (2010). *SilentEye - Steganography is yours*. Retrieved August 28, 2012, from <http://www.silenteye.org/?referer=app>
- Constine, J. (2010). Facebook announces seamless messaging across communication mediums (Inside Facebook). Retrieved December 30, 2012, from <http://www.insidefacebook.com/2010/11/15/seamless-messaging-communication-medium/>
- Cosic, J., & Baca, M. (2010). Steganography and steganalysis - Does local web sites contain “Stego” contents? *52nd International Symposium ELMAR-2010*. 85–88. Zadar, Croatia: IEEE. Retrieved from <http://ieeexplore.ieee.org.ezproxy.aut.ac.nz/stamp/stamp.jsp?tp=&arnumber=5606088&isnumber=5606063>
- Craiger, J. (2006). Computer forensics procedures and methods. To appear in H. Bigdoli (Ed.), *Handbook of Information Security*. John Wiley & Sons. Retrieved from <http://ncfs.org/craiger.forensics.methods.procedures.final.pdf>
- Curran, K., & Devitt, J. M. (2008). Image analysis for online dynamic steganography detection. *Computer and Information Science*, 1(3), 32–41. Retrieved from <http://ccsenet.org/journal/index.php/cis/article/viewFile/1825/1735..>
- Das, S., Das, S., Bandyopadhyay, B., & Sanyal, S. (2011). Steganography and steganalysis: different approaches. *Cornell University Library*. Retrieved March 12, 2012, from <http://arxiv.org/abs/1111/3758>
- Dunbar, B. (2002). A detailed look at steganographic techniques and their use in an open-systems environment. *SANS Information Security*

Reading Room. Retrieved March 12, 2012 from http://www.sans.org/reading_room/whitepapers/covert/detailed-steganographic-techniques-open-systems-environment_677

- Engle, S. (2003). *Current state of steganography: Uses, limits, & implications*. Retrieved March 12, 2012, from <https://sites.google.com/a/ucdavis.edu/sjengle/Research/state-of-steganography>
- Freeman, K. (2012). Facebook Rolls Out File-Sharing for All Groups [EXCLUSIVE]. *Mashable*. Retrieved August 27, 2012, from <http://mashable.com/2012/05/10/Facebook-groups-3/>
- Fridrich, J. (2010). *Steganography in digital media*. Cambridge, UK: Cambridge University Press.
- Hamid, N., Yahya, A., Ahmad, R. B., & Al-Qershi, O. M. (2012). Image steganography techniques: An overview. *International Journal of Computer Science and Security (IJCSS)*, 6(3), 168–187. Retrieved from <http://www.cscjournals.org/csc/manuscript/Journals/IJCSS/volume6/Issue3/IJCSS-670.pdf>
- Hani. (2009). Empirical research. *Explorable*. Retrieved October 11, 2012, from <http://www.experiment-resources.com/empirical-research.html>
- Hayati, P., Potdar, V., & Chang, E. (2007). *A survey of steganographic and steganalytic tools for the digital forensic investigator*. Retrieved from http://www.pedramhayati.com/images/docs/survey_of_steganography_and_steganalytic_tools.pdf
- Hayes, G. (2011). Social media used for criminal investigations. *The Record*. Retrieved April 17, 2012, from http://therecordlive.com/article/Orange_County_News/Orange_County_News/Social_media_used_for_criminal_investigations/64391

- Hosmer, C. (2006). Discovering hidden evidence. *Journal of Digital Forensic Practice*, 1(1), 47–56. doi:10.1080/15567280500541447
- Hosmer, C., & Hyde, C. (2003). Discovering covert digital evidence. *Digital Forensic Research Workshop* (pp. 1–5). Cleveland, Ohio. Retrieved from <http://www.dfrws.org/2003/presentations/Paper-Hosmer-digitalevidence.pdf>
- Huber, M., Schrittwieser, S., Mulazzani, M., Wondracek, G., Leithner, M., & Weippl, E. (2011). Social snapshots: Digital forensics for online social networks. *Annual Computer Security Applications Conference (ACSAC)*. 113-122. doi: 10.1145/2076732.2076748
- Huge rise in social media “crimes”. (2012, December 27). *BBC News*. Retrieved from <http://www.bbc.co.uk/news/uk-20851797>
- Ibrahim, A. (2007). *Steganalysis in computer forensics*. Proceedings of the 5th Australian Digital Forensics Conference, Edith Cowan University, Perth, Australia. Retrieved from <http://ro.ecu.edu.au/cgi/viewcontent.cgi?article=1009&context=adf>
- INFOAVE. (2011). *Your guide to common Web image formats*. Retrieved June 8, 2012, from <http://thundercloud.net/infoave/new/2011/04/10/your-guide-to-common-web-image-formats/#.UFqwgLLiaLx>
- Johnson, N. F., & Jajodia, S. (1998). Exploring steganography: Seeing the unseen. *Computing Practices*, 31(2), 26–34. doi:10.1109/MC.1998.4655281
- Jones, K. J., & Belani, R. (2010a). Web browser forensics, Part 1. *Symantec Connect*. Retrieved July 18, 2012, from <http://www.symantec.com/connect/articles/web-browser-forensics-part-1>
- Jones, K. J., & Belani, R. (2010b). Web browser forensics, Part 2. *Symantec Connect*. Retrieved July 18, 2012, from

<http://www.symantec.com/connect/articles/web-browser-forensics-part-2>

- Kent, K., Chevalier, S., Grance, T., & Dang, H. (2006). Guide to integrating forensic techniques into incident response. *NIST Special Publication*. Retrieved from <http://csrc.nist.gov/publications/nistpubs/800-86/SP800-86.pdf>
- Kessler, G. C. (2004a). An overview of steganography for the computer forensics examiner. *Forensic Science Communications*. Retrieved March 2, 2012, from http://www.fbi.gov/about-us/lab/forensic-science-communications/fsc/july2004/research/2004_03_research01.htm
- Kessler, G. C. (2004b). Steganography: Implications for the prosecutor and computer forensics examiner. *Child Sexual Exploitation Program Newsletter Archives*, 1(1). Retrieved from http://www.ndaa.org/pdf/Update_gr_v1_no1.pdf
- Kipper, G. (2004). *Investigator's guide to steganography*. Boca Raton, Florida: CRC Press LLC.
- Kohn, M., Eloff, J., & Olivier, M. (2006). Framework for a digital forensic investigation. *Proceedings of the ISSA 2006 from Insight to Foresight Conference*. Retrieved from http://icsa.cs.up.ac.za/issa/2006/Proceedings/Full/101_Paper.pdf
- Kumar, A., & Pooja, K. (2010). Steganography - A data hiding technique. *International Journal of Computer Applications*, 9(7), 19–23. doi:10.5120/1398-1887
- List of social networking websites. (2013). Retrieved March 1, 2013, from Wikipedia, http://en.wikipedia.org/wiki/List_of_social_networking_websites

- Malik, H. (2009). Critical analysis of digital steganography. In S. Lian & Y. Zhang (Eds.), *Handbook of research on secure multimedia distribution* (pp. 352–382). Hershey, PA: IGI Global.
doi:10.4018/978-1-60566-262-6.ch019
- McKemmish, R. (1999). What is forensic computing? *Trends and Issues in Crime and Criminal justice*, (June). Retrieved from
<http://aic.gov.au/documents/9/C/A/%7B9CA41AE8-EADB-4BBF-9894-64E0DF87BDF7%7Dt118.pdf>
- Messages basics (n.d.) - in *Facebook help center*. Retrieved September 2, 2012, from <https://www.Facebook.com/help/messages/basics>
- Microsoft Support. (2010). *RAM, virtual memory, pagefile, and memory management in Windows*. Retrieved August 12, 2012, from
<http://support.microsoft.com/kb/2160852>
- Morkel, T., Eloff, J., & Olivier, M. (2005). An overview of image steganography. *Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA2005)*. Retrieved from
http://icsa.cs.up.ac.za/issa/2005/Proceedings/Full/098_Article.pdf
- Morsy, H. A., Nossair, Z. B., Hamdy, A. M., & Amer, F. Z. (2011). Information hiding by inverting the LSB bits of DCT coefficients of JPEG images. *Journal of American Science*, 7(11), 171–177.
Retrieved from http://www.jofamericanscience.org/journals/am-sci/am0711/020_7282am0711_171_177.pdf
- Mostyn, S. (2010). Police stats suggest Facebook becoming hotbed of crime. *The Tech Herald*. Retrieved April 25, 2012, from
<http://www.thetechherald.com/articles/Police-stats-suggest-Facebook-becoming-hotbed-of-crime>
- Mulazzani, M., Huber, M., & Weippl, E. (2012). Social network forensics: Tapping the data pool of social networks. *Eighth Annual IFIP WG*

11.9 International Conference on Digital Forensics. Retrieved from http://www.sba-research.org/wp-content/uploads/publications/socialForensics_preprint.pdf

- Munoz, A. (2007). *StegSecret. A simple steganalysis tool*. Retrieved August 17, 2012, from <http://stegsecret.sourceforge.net/>
- Mutawa, N. Al Awadhi, I. Al Baggili, I., & Marrington, A. (2011). Forensic artefacts of Facebook's instant messaging service. *6th International Conference on Internet Technology and Secured Transactions*, 771–776. Abu Dhabi, UAE: IEEE. Retrieved from <http://ieeexplore.ieee.org.ezproxy.aut.ac.nz/stamp/stamp.jsp?tp=&arnumber=6148436&isnumber=6148349>
- National Institute of Justice. (2004). *NIJ: Special Report – Forensic examination of digital evidence: A guide for law enforcement*. Retrieved from <https://www.ncjrs.gov/pdffiles1/nij/199408.pdf>
- National Institute of Justice. (2010). *Digital Evidence Analysis: Steganography Detection*. Retrieved 20 July, 2011, from <http://www.nij.gov/topics/forensics/evidence/digital/analysis/steganography.htm>
- Newman, R. C. (2007). Covert computer and network communications. In *Proceedings of the 4th annual conference on Information security curriculum development (InfoSecCD '07)*. NY, USA: ACM. doi:10.1145/1409908.1409922
- Noureldin, S. H., Hashem, S., & Abdalla, S. (2011). Computer Forensics Guidance Model with Cases Study. *2011 Third International Conference on Multimedia Information Networking and Security*, 564–571. doi:10.1109/MINES.2011.49

- Oh, J., Lee, S., & Lee, S. (2011). Advanced evidence collection and analysis of web browser activity. *Digital Investigation*, 8, S62–S70. doi:10.1016/j.diin.2011.05.008
- Oh, J., Son, N., Lee, S., & Lee, K. (2012). A study for classification of web browser log and timeline visualization. *The 13th International Workshop on Information Security Applications (WISA2012)*. Retrieved from [http://isaa.sch.ac.kr/wisa2012/%EB%85%BC%EB%AC%B8/Session 4/4-201_A Study for Classification of Web Browser Log and Timeline Visualization.pdf](http://isaa.sch.ac.kr/wisa2012/%EB%85%BC%EB%AC%B8/Session%204/4-201_A%20Study%20for%20Classification%20of%20Web%20Browser%20Log%20and%20Timeline%20Visualization.pdf)
- Patzakis, J. (2011). Facebook evidence disallowed by court due to lack of “Identifying Characteristics”. *eDiscovery Law & Tech Blog*. Retrieved August 15, 2012, from <http://blog.x1discovery.com/2011/10/03/Facebook-evidence-disallowed-by-court-due-to-lack-of-identifying-characteristics/>
- Patzakis, J. (2012). 689 Published cases involving social media evidence. *eDiscovery Law & Tech Blog*. Retrieved August 15, 2012, from <http://blog.x1discovery.com/2012/03/14/689-published-cases-involving-social-media-evidence-with-full-case-listing/>
- Pereira, M. T. (2009). Forensic analysis of the Firefox 3 Internet history and recovery of deleted SQLite records. *Digital Investigation*, 5(3-4), 93–103. doi:10.1016/j.diin.2009.01.003
- Pollitt, M. M. (1995). Computer forensics: An approach to evidence in cyberspace. *National Information System Security Conference*, 2, 487–491. Retrieved from <http://www.digitalevidencepro.com/Resources/Approach.pdf>
- Por, L. Y., & Delina, B. (2008). Information hiding: A new approach in text steganography. *7th WSEAS International Conference on Applied Computer & Applied Computational Science (ACACOS)*, 689–695.

Retrieved from <http://www.wseas.us/e-library/conferences/2008/hangzhou/acacos/116-586-634.pdf>

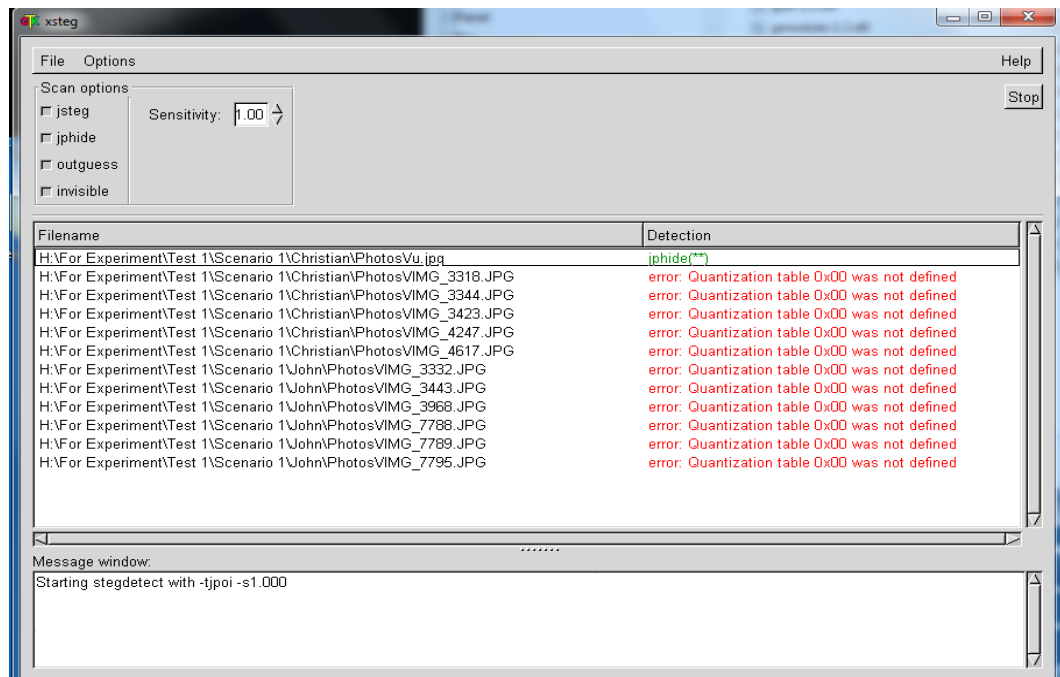
- Potdar, V. M., Khan, M. A., Chang, E., Ulieru, M., & Worthington, P. R. (2005). e-Forensics steganography system for secret information retrieval. *Advanced Engineering Informatics*, 19(3), 235–241. doi:10.1016/j.aei.2005.04.003
- Provos, N., & Honeyman, P. (2001). *Detecting steganographic content on the internet*. In *Network and Distributed System Security Symposium (ISOC NDSS'02)*. Retrieved from <http://www.citi.umich.edu/techreports/reports/citi-tr-01-11.pdf>
- Raphael, A. J., & Sundaram, V. (2011). Cryptography and steganography – A survey. *International Journal of Computer Technology and Applications*, 2(3), 626–630. Retrieved from <http://www.ijcta.com/documents/volumes/vol2issue3/ijcta2011020338.pdf>
- Reith, M., Carr, C., & Gunsch, G. (2002). An examination of digital forensic models. *International Journal of Digital Evidence*, 1(3), 1–12. Retrieved from http://people.umich.edu/pstephen/other_papers/Digital_Forensic_Models.pdf
- Retrieving digital evidence: Methods, techniques, and issues: Part 3. (2012). *Digital Forensic Investigator News*. Retrieved August 12, 2012, from <http://www.dfinews.com/article/retrieving-digital-evidence-methods-techniques-and-issues-part-3>
- Ruotolo, J. (2012). Social media sleuthing: Proceeding with caution in the new frontier. *Property casualty 360*. Retrieved 20 March, 2013, from <http://www.propertycasualty360.com/2012/01/27/social-media-sleuthing>

- Savoldi, A., Gubian, P., & Echizen, I. (2010). Uncertainty in live forensics. In K. Chow & S. Shenoi (Eds.), *Advances in Digital Forensics VI* (pp. 171–184). Berlin, Germany: Springer Berlin Heidelberg.
- Scoville, D. (2011). Social media: Online investigation. *Police the law enforcement magazine*. Retrieved April 17, 2012, from www.policemag.com/Channel/Technology/Articles/2011/10/Online-Investigation.aspx
- Sheetz, M. W. (2003). Reading between the lines: Steganography. *Law & Order*, 51(12), 46–51. Retrieved from <http://ezproxy.aut.ac.nz/login?url=http://search.proquest.com/docview/197219230?accountid=8440>
- Sutherland, I., Evans, J., Tryfonas, T., & Blyth, A. (2008). Acquiring volatile operating system data tolls and techniques. *ACM SIGOPS Operating Systems Review*, 42(3), 65–73. doi:10.1145/1368506.1368516
- Tone, D. (2012). Backbone security annouces enhanced steganography detection tool. *Send2Press Newswire*. Retrieved September 20, 2012, from <http://www.send2press.com/newswire/2012-02-0208-003.shtml>
- Trapani, G. (2007). *Geek to live: Hide data in files with easy steganography tools*. Retrieved August 4, 2012, from <http://lifehacker.com/230915/geek-to-live--hide-data-in-files-with-easy-steganography-tools>
- Wade, A. (2012, September 12). Criminals target social media. *The New Zealand Herald*. Retrieved from http://www.nzherald.co.nz/nz/news/article.cfm?c_id=1&objectid=10833397

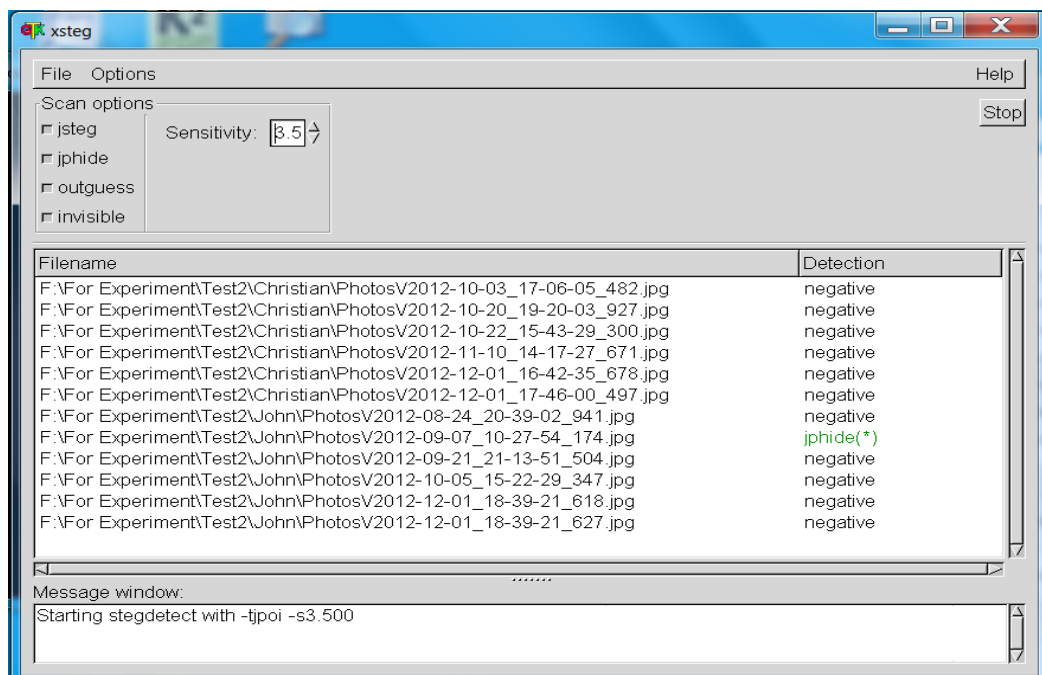
- Wong, K., Lai, A. C. T., Yeung, J. C. K., Lee, W. L., & Chan, P. H. (2011). Facebook forensics. *Valkyrie-X Security Research Group*. Retrieved from http://hackveda.vmdtech.org/pdf/Facebook_Forensics-Finalized.pdf
- Zainudin, N., Merabti, M., & Llewellyn-Jones, D. (2010). *A digital forensic investigation model for online social networking*. Paper presented at 11th Annual Postgraduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting, Liverpool, UK. Retrieved from <http://www.cms.livjm.ac.uk/pgnet2010/MakeCD/Papers/2010042.pdf>
- Zainudin, N., Merabti, M., & Llewellyn-Jones, D. (2011). Online social networks as supporting evidence: A digital forensic investigation model and its application design. *International Conference on Research and Innovation in Information System (ICRIIS)*, 1-6. doi: 10.1109/ICRIIS.2011.6125728
- Zax, R., & Adelstein, F. (2009). FAUST: Forensic artefacts of uninstalled steganography tools. *Digital Investigation*, 6(1-2), 25–38. doi:10.1016/j.diin.2009.02.002
- Zip (file format). (2012). *In Wikipedia*. Retrieved August 5, 2012, from [http://en.wikipedia.org/wiki/Zip_\(file_format\)](http://en.wikipedia.org/wiki/Zip_(file_format))

APPENDICES

Appendix 1 –Possible Errors from StegDetect



Appendix 2 – Scenario 1 Experimental Images before JP Hide and Seek Steganographic Process (1 False Positive)



Appendix 3 - Scenario 1: Christian Riley's Imaged Hard Drive Verification

Created By AccessData® FTK® Imager 3.0.1.1467 110406

Case Information:

Acquired using: ADI3.0.1.1467

Case Number: Scenario 1

Evidence Number: 001

Unique description: CRiley_T2

Examiner: Aimie Chee

Notes: Target: Christian Riley

Information for H:\MFIT Thesis Experiment Data Collection\Test
2\Christian\HDD Image File\CRiley_Test2:

Physical Evidentiary Item (Source) Information:

[Drive Geometry]

Cylinders: 14,593

Tracks per Cylinder: 255

Sectors per Track: 63

Bytes per Sector: 512

Sector Count: 234,441,648

[Physical Drive Information]

Drive Serial Number: WD-WXCZ07003402

Drive Interface Type: USB

Source data size: 114473 MB

Sector count: 234441648

[Computed Hashes]

MD5 checksum: 3f4eeee22c698b9f74b3a7fa783f8b43

SHA1 checksum: b01f51a8ff371bb3c039457c898cc49299e4bc7d

Image Information:

Acquisition started: Mon Dec 10 13:13:15 2012

Acquisition finished: Mon Dec 10 15:00:49 2012

Segment list:

H:\MFIT Thesis Experiment Data Collection\Test 2\Christian\HDD Image
File\CRiley_Test2.E01

H:\MFIT Thesis Experiment Data Collection\Test 2\Christian\HDD Image
File\CRiley_Test2.E02

H:\MFIT Thesis Experiment Data Collection\Test 2\Christian\HDD Image
File\CRiley_Test2.E03

H:\MFIT Thesis Experiment Data Collection\Test 2\Christian\HDD Image
File\CRiley_Test2.E04

Image Verification Results:

Verification started: Mon Dec 10 15:00:50 2012

Verification finished: Mon Dec 10 15:24:20 2012

MD5 checksum: 3f4eeee22c698b9f74b3a7fa783f8b43 : verified

SHA1 checksum: b01f51a8ff371bb3c039457c898cc49299e4bc7d : verified

Appendix 4 - Scenario 1: John Doe's Imaged Hard Drive Verification

Created By AccessData® FTK® Imager 3.0.1.1467 110406

Case Information:

Acquired using: ADI3.0.1.1467

Case Number: Scenario 1

Evidence Number: 004

Unique description: Terrorism Related Case

Examiner: Aimie Chee

Notes: Target: John Doe

Information for H:\MFIT Thesis Experiment Data Collection\Test 2\John\HDD
Acquisition\JDoe_Test2:

Physical Evidentiary Item (Source) Information:

[Drive Geometry]

Cylinders: 14,593

Tracks per Cylinder: 255

Sectors per Track: 63

Bytes per Sector: 512

Sector Count: 234,441,648

[Physical Drive Information]

Drive Model: WD 1200BEV External USB Device

Drive Serial Number: WD-WXEZ07L46465

Drive Interface Type: USB

Source data size: 114473 MB

Sector count: 234441648

[Computed Hashes]

MD5 checksum: 958bd515e4cf39e350cbf49396724832

SHA1 checksum: be34dbd4d0d0a086d840b40c8b7cfcaa50b11900

Image Information:

Acquisition started: Mon Dec 10 16:38:09 2012

Acquisition finished: Mon Dec 10 18:49:59 2012

Segment list:

H:\MFIT Thesis Experiment Data Collection\Test 2\John\HDD
Acquisition\JDoe_Test2.E01

H:\MFIT Thesis Experiment Data Collection\Test 2\John\HDD
Acquisition\JDoe_Test2.E02

H:\MFIT Thesis Experiment Data Collection\Test 2\John\HDD
Acquisition\JDoe_Test2.E03

Image Verification Results:

Verification started: Mon Dec 10 18:50:01 2012

Verification finished: Mon Dec 10 19:27:22 2012

MD5 checksum: 958bd515e4cf39e350cbf49396724832 : verified

SHA1 checksum: be34dbd4d0d0a086d840b40c8b7cfcaa50b11900 : verified

Appendix 5 - Scenario 1: Christian Riley's Imaged RAM Verification

Created By AccessData® FTK® Imager 3.0.1.1467 110406

Case Information:

Acquired using: ADI3.0.1.1467

Case Number: Scenario 1

Evidence Number: 002

Unique description: Memory Dump

Examiner: Aimie Chee

Notes: Memory Dump from Target: Chritian Riley

Information for C:\Users\AIMIE CHEE\Desktop\Test 2 memory
Acquisition\CRiley\Live memory image\Test2_liveMemory_cRiley:

Physical Evidentiary Item (Source) Information:

[Drive Geometry]

Bytes per Sector: 512

Sector Count: 6,023,168

[Image]

Image Type: Raw (dd)

Source data size: 2941 MB

Sector count: 6023168

[Computed Hashes]

MD5 checksum: 0a3adb61abada31c8642b92ef3e3c25f

SHA1 checksum: 1a4be302d5169726d32090694fbde666ae2aa9d8

Image Information:

Acquisition started: Mon Dec 10 15:38:29 2012

Acquisition finished: Mon Dec 10 15:40:34 2012

Segment list:

C:\Users\AIMIE CHEE\Desktop\Test 2 memory Acquisition\CRiley\Live
memory image\Test2_liveMemory_cRiley.E01

Image Verification Results:

Verification started: Mon Dec 10 15:40:35 2012

Verification finished: Mon Dec 10 15:42:24 2012

MD5 checksum: 0a3adb61abada31c8642b92ef3e3c25f : verified

SHA1 checksum: 1a4be302d5169726d32090694fbde666ae2aa9d8 : verified

Appendix 6 - Scenario 1: John Doe's Imaged RAM Verification

Created By AccessData® FTK® Imager 3.0.1.1467 110406

Case Information:

Acquired using: ADI3.0.1.1467

Case Number: Scenario 1

Evidence Number: 003

Unique description: Memory Dump

Examiner: Aimie Chee

Notes: Memory Dump from target: John Doe

Information for C:\Users\AIMIE CHEE\Desktop\Test 2 memory
Acquisition\JDoe\acquisition on memory file\Test2_liveMemory_JDoe:

Physical Evidentiary Item (Source) Information:

[Drive Geometry]

Bytes per Sector: 512

Sector Count: 4,190,208

[Image]

Image Type: Raw (dd)

Source data size: 2046 MB

Sector count: 4190208

[Computed Hashes]

MD5 checksum: fe0106baf77a666ab23119aeff2c71d5

SHA1 checksum: 81cc54a1e9f5f6483d88de0624121ca23182ca42

Image Information:

Acquisition started: Mon Dec 10 16:00:44 2012

Acquisition finished: Mon Dec 10 16:02:38 2012

Segment list:

C:\Users\AIMIE CHEE\Desktop\Test 2 memory Acquisition\JDoe\acquisition
on memory file\Test2_liveMemory_JDoe.E01

Image Verification Results:

Verification started: Mon Dec 10 16:02:38 2012

Verification finished: Mon Dec 10 16:04:00 2012

MD5 checksum: fe0106baf77a666ab23119aeff2c71d5 : verified

SHA1 checksum: 81cc54a1e9f5f6483d88de0624121ca23182ca42 : verified

Appendix 7 – Scenario 2: John Doe’s Imaged Hard Drive Verification

Created By AccessData® FTK® Imager 3.0.1.1467 110406

Case Information:

Acquired using: ADI3.0.1.1467

Case Number: Scenario2

Evidence Number: 01

Unique description: John01

Examiner: Aimie Chee

Notes:

Information for C:\MFIT Thesis Experiment Data Collection\Scenario2\S2:

Physical Evidentiary Item (Source) Information:

[Drive Geometry]

Cylinders: 19,457

Tracks per Cylinder: 255

Sectors per Track: 63

Bytes per Sector: 512

Sector Count: 312,581,808

[Physical Drive Information]

Drive Model: WD 1600BEV External USB Device

Drive Serial Number: WD-WXEZ07A58058

Drive Interface Type: USB

Source data size: 152627 MB

Sector count: 312581808

[Computed Hashes]

MD5 checksum: 18a5fe4bc214d8fe4f8be219f4283273

SHA1 checksum: 5d5ff71e23ee17ce62bf35a264e59a72ba9a0c50

Image Information:

Acquisition started: Sat Nov 17 21:13:14 2012

Acquisition finished: Sat Nov 17 23:35:43 2012

Segment list:

C:\MFIT Thesis Experiment Data Collection\Scenario2\S2.E01

C:\MFIT Thesis Experiment Data Collection\Scenario2\S2.E02

C:\MFIT Thesis Experiment Data Collection\Scenario2\S2.E03

C:\MFIT Thesis Experiment Data Collection\Scenario2\S2.E04

Image Verification Results:

Verification started: Sat Nov 17 23:35:47 2012

Verification finished: Sat Nov 17 23:56:23 2012

MD5 checksum: 18a5fe4bc214d8fe4f8be219f4283273 : verified

SHA1 checksum: 5d5ff71e23ee17ce62bf35a264e59a72ba9a0c50 : verified

Appendix 8 – Facebook Pre-Test Photo Identifier

Facebook (FB)						
FB Features	JP Hide and Seek (JPEG)	SilentEye (JPEG)	End of File (JPEG)	StegHide (BMP)	S-Tools (GIF)	Invisible Secrets 4 (PNG)
Photo Upload	FB_P1	FB_P2	FB_P3	FB_P4	FB_P5	FB_P6
File Sharing	FB_P7	FB_P8	FB_P9	FB_P10	FB_P11	FB_P12
Message Attachment	FB_P13	FB_P14	FB_P15	FB_P16	FB_P17	FB_P18

Appendix 9 – Google+ Pre-Test Photo Identifier

Google+						
Google+ Features	JP Hide and Seek (JPEG)	SilentEye (JPEG)	End of File (JPEG)	StegHide (BMP)	S-Tools (GIF)	Invisible Secrets 4 (PNG)
Photo Upload	G_P1	G_P2	G_P3	G_P4	G_P5	G_P6
File Sharing	N/A	N/A	N/A	N/A	N/A	N/A
Message Attachment	N/A	N/A	N/A	N/A	N/A	N/A

Appendix 10 – Facebook Pre-Test Configuration and Results

Facebook											
Features	Tools used	Image Format Used	Cover Image File Name	Cover Image MD5	Stego Image File Name	Stego Image MD5	Downloaded Image File Name	Downloaded Image MD5	Successful Secret Message Extraction		Remarks
									Yes	No	
Photo Upload	JP Hide and Seek	JPEG	FB_P1	3CB285FF1B0676EAA800CA749E7F8051	SFB_P1	F291AB533E38361937CB2900962C6481	396290_168468496625192_207093110_n.jpg	DC6CAF38A4A58A4A1450BFDC31ED1080		X	Secret Message: secret.txt Hidden Message Extraction: Unable to extract secret message. The tool showed wrong passphrase although the right passphrase was used for extraction
	Silent Eye	JPEG	FB_P2.	A57F62DCF4C0BCE70F4C6937D0DE5C1D	SFB_P2	7123502890684D2218F38991B2122070	149889_168496316622410_84868167_n.jpg	BAB0DD336E512777A1FEE8264412128A		X	Luminance Interval = 5 JPG Quality = 30% Header position = bottom CharSet = ASCII Secret Message: Officials in many states said they have difficulties detecting forged birth certificates. Verifying date of birth is also required by the Act, and a system exists for doing so, but no licensing agencies are using it because of concerns about incomplete data, among other reasons. Partly because these two systems are not fully operational, GAO investigators were able to use counterfeit out-of-state drivers' licenses and birth certificates to fraudulently obtain licenses in three states. Secret Message Extraction: Unable to reveal the secret message, tool showed a message "The media don't seem to have a hidden message".

Facebook											
Features	Tools used	Image Format Used	Cover Image File Name	Cover Image MD5	Stego Image File Name	Stego Image MD5	Downloaded Image File Name	Downloaded Image MD5	Successful Secret Message Extraction		Remarks
									Yes	No	
	Silent Eye	JPEG	FB_P2	A57F62DCF4 C0BCE70F4 C6937D0DE5 C1D	SFB_P2_10	4403D48C8A9D D4B3C8E33127 EF1F2304	17927_10200273 659509009_1013 802319_n.jpg	64C7508E21DC2 8AFC30F6F2F78 880C3A	x		Luminance Internal = 10 JPG Quality = 30% Header Position = bottom CharSet = ASCII Secret Message: Officials in many states said they have difficulties detecting forged birth certificates. Verifying date of birth is also required by the Act, and a system exists for doing so, but no licensing agencies are using it because of concerns about incomplete data, among other reasons. Partly because these two systems are not fully operational, GAO investigators were able to use counterfeit out-of-state drivers' licenses and birth certificates to fraudulently obtain licenses in three states. Hidden Message Extraction: It was successfully extracted
	EOF	JPEG	FB_P3	9F62377388B CCB21F914C 187AA94B00 4	SFB_P3	DB4E1A6DAD 86584C921B97 F652A4DC8C	65363_16846883 3291825_210205 8775_n.jpg	A74E5B9D73775 3C98C590B39B6 53160A		x	Secret message: The GAO's investigators obtained five driver's licenses in three different states under fictitious identities using combinations of names, birthdates, and Social Security Numbers together with counterfeit documents. In two states, a GAO investigator was able to obtain two licenses with different identities using the same person's face. Only in one case did a motor vehicle employee appear to question the validity of the documents being presented—but the GAO investigator was still able to obtain a driver's license. “The investigators exploited cross state information vulnerabilities by using counterfeit

Facebook											
Features	Tools used	Image Format Used	Cover Image File Name	Cover Image MD5	Stego Image File Name	Stego Image MD5	Downloaded Image File Name	Downloaded Image MD5	Successful Secret Message Extraction		Remarks
									Yes	No	
											driver’s licenses from states other than the ones where they were issued valid driver’s licenses. There’s nothing to stop tomorrow’s terrorists from obtaining valid driver’s licenses in states where a pre-9/11 mentality prioritizes speedy customer service over the careful identity authentication of applicants. There are technological solutions to counter this risk, some of those made possible by federal systems and some from industry. The REAL ID Act identity security standards published in 2008 provide a framework for closing vulnerabilities like those exploited by GAO investigators, but compliance with those rules is voluntary,” Zimmer noted. Hidden Message Extraction: When the downloaded picture was opened with the Notepad application, the inserted secret message was not readable
	StegHide	BMP	FB_P4	1F531100D97866306F0396E6C94B0969	SFB_P4	671C3ED6CD1BAE108F243541A2EB9612	314295_168469123291796_485452816_n.jpg	2C4B33B55B892F8608F75949399B5E90		X	Encryption = Rijndael-128 Mode = cbc Compression = 9 Secret Message = secret.txt Hidden Message Extraction: During extraction from the downloaded image, the hidden data was unable to be extracted and the tool showed invalid password, although the password used was correct. Moreover, the file type has been changed from .bmp to .jpg by Facebook.

Facebook											
Features	Tools used	Image Format Used	Cover Image File Name	Cover Image MD5	Stego Image File Name	Stego Image MD5	Downloaded Image File Name	Downloaded Image MD5	Successful Secret Message Extraction		Remarks
									Yes	No	
	S-Tools	GIF	FB_P5	BBC998385A9D929E64A4183814B271DB	SFB_P5	FA6DDB962C614974599854E02FC53D92	374155_168469276625114_1388874899_n.jpg	4D2EF8FA480F23AE293EC89EDEEE7196		x	Encryption = IDEA Secret Message = secret.txt Secret Message Extraction: STool unable to open the .jpg file and thus unable to reveal the secret message. Image format has been changed from .gif to .jpg by Facebook automatically.
	Invisible Secrets 4	PNG	FB_P6	6334EB373E278B9BDA5026A75AA2AA59	SFB_P6	36790A0A4DF51DE51E5E900D7CB5F402	527476_168469503291758_1816077072_n.jpg	0A483E499229C73E1C0BB66880DFBA68		x	Encryption = AES Rijndael Secret Message = secret.txt Secret Message Extraction: Unable to extract. Warning message by the tool has showed - "Access Denied! Invalid carrier file, password or algorithm". Image format has changed automatically by Facebook.
File Sharing	JP Hide and Seek	JPEG	FB_P7	8D89E4D1A371DDD5F2C64ACC84C87C56	SFB_P7	1D205DCE1F190A3ADB0CE768AB305289	SFB_P7.jpg	1D205DCE1F190A3ADB0CE768AB305289	x		Secret Message: secret.txt Secret Message Extraction: Successful

Facebook											
Features	Tools used	Image Format Used	Cover Image File Name	Cover Image MD5	Stego Image File Name	Stego Image MD5	Downloaded Image File Name	Downloaded Image MD5	Successful Secret Message Extraction		Remarks
									Yes	No	
	Silent Eye	JPEG	FB_P8	4ECFC5F946CF9B40989C02246EE6D777	SFB_P8	07491B6BC8406FDB8C8CA57353882295	SFB_P8.jpg	07491B6BC8406FDB8C8CA57353882295	x		Luminance Interval = 5 JPG Quality = 30% Header position = bottom CharSet = ASCII Secret Message: Officials in many states said they have difficulties detecting forged birth certificates. Verifying date of birth is also required by the Act, and a system exists for doing so, but no licensing agencies are using it because of concerns about incomplete data, among other reasons. Partly because these two systems are not fully operational, GAO investigators were able to use counterfeit out-of-state drivers' licenses and birth certificates to fraudulently obtain licenses in three states. Secret Message Extraction: Successful
	EOF	JPEG	FB_P9	B6CEBA95C3776B0FD4233E75D3F3B500	SFB_P9	3360374BAB6843192E95759BBFBA6C42	SFB_P9.jpg	3360374BAB6843192E95759BBFBA6C42	x		Secret Message: Same as FB_P3 Secret Message Extraction: When the downloaded picture was opened with the Notepad application, the appended secret message is readable
	StegHide	BMP	FB_P10	25FA1F050C93D082199A2839C1B64D7B	SFB_P10	D5BCE650764AA64D80FA2BBB2D98E6	SFB_P10.bmp	D5BCE650764AA64D80FA2BEBB2D98E6	x		Encryption = Rijndael-128 Mode = cbc Compression = 9 Secret Message = secret.txt Secret Message Extraction: Successful

Facebook											
Features	Tools used	Image Format Used	Cover Image File Name	Cover Image MD5	Stego Image File Name	Stego Image MD5	Downloaded Image File Name	Downloaded Image MD5	Successful Secret Message Extraction		Remarks
									Yes	No	
	S-Tools	GIF	FB_P11	19CFA1BAB8C793063E87442481AFAEE0	SFB_P11	8912D4AD8503701C0D55B1A7054C4278	SFB_P11.gif	8912D4AD8503701C0D55B1A7054C4278	x		Encryption = IDEA Secret Message = secret.txt Secret Message Extraction: Successful
Message Attachment	Invisible Secrets 4	PNG	FB_P12	A9E3CC8787695E127999E566402AD7F0	SFB_P12	F621A70D80052F3241AAECF7CA204C3A	SFB_P12.png	F621A70D80052F3241AAECF7CA204C3A	x		Encryption = AES Rijndael Secret Message = secret.txt Secret Message Extraction: Successful
	JP Hide and Seek	JPEG	FB_P13	BE8C5CCD4928A9362978F1103CE8C499	SFB_P13	6425EEF9E338840B1C622B4A80CED67C	SFB_P13.jpg	6425EEF9E338840B1C622B4A80CED67C	x		Secret Message: secret.txt Secret Message Extraction: Successful

Facebook											
Features	Tools used	Image Format Used	Cover Image File Name	Cover Image MD5	Stego Image File Name	Stego Image MD5	Downloaded Image File Name	Downloaded Image MD5	Successful Secret Message Extraction		Remarks
									Yes	No	
	Silent Eye	JPEG	FB_P14	83283920B79698CE2C1CB1809E425A69	SFB_P14	466EDF8FEBCC5A7450B71F6F8656D256	SFB_P14.jpg	466EDF8FEBCC5A7450B71F6F8656D256	x		Luminance Interval = 5 JPG Quality = 30% Header position = bottom CharSet = ASCII Secret Message: Officials in many states said they have difficulties detecting forged birth certificates. Verifying date of birth is also required by the Act, and a system exists for doing so, but no licensing agencies are using it because of concerns about incomplete data, among other reasons. Partly because these two systems are not fully operational, GAO investigators were able to use counterfeit out-of-state drivers' licenses and birth certificates to fraudulently obtain licenses in three states. Secret Message Extraction: Successful
	EOF	JPEG	FB_P15	7814F3DF301AA9B178E1379F1A0BC760	SFB_P15	9671F014EA8805AA3E5565DA09F7EB2C	SFB_P15.jpg	9671F014EA8805AA3E5565DA09F7EB2C	x		Secret Message: Same as FB_P3 Secret Message Extraction: When the downloaded picture was opened with Notepad application, the appended secret message is readable
	StegHide	BMP	FB_P16	D534861A146A8A1832822D6E4007CA54	SFB_P16	3930406ACAE DCE7F258A3129E2E31E74	SFB_P16.bmp	3930406ACAEDCE7F258A3129E2E31E74	x		Encryption = Rijndael-128 Mode = cbc Compression = 9 Secret Message = secret.txt Secret Message Extraction: Successful

Facebook											
Features	Tools used	Image Format Used	Cover Image File Name	Cover Image MD5	Stego Image File Name	Stego Image MD5	Downloaded Image File Name	Downloaded Image MD5	Successful Secret Message Extraction		Remarks
									Yes	No	
	S-Tools	GIF	FB_P17	11B6BA272E35E9264D677F96B31FF2C2	SFB_P17	150A3249D46B14B771FFB21F8A22EABF	SFB_P17.gif	150A3249D46B14B771FFB21F8A22EABF	x		Encryption = IDEA Secret Message = secret.txt Secret Message Extraction: Successful
	Invisible Secrest 4	PNG	FB_P18	4B698CAFC EF30D71E176109C3FC17F27	SFB_P18	71121E88E1A16F083F170C64048787F9	SFB_P18.png	71121E88E1A16F083F170C64048787F9	x		Encryption = AES Rijndael Secret Message = secret.txt Secret Message Extraction: Successful

Appendix 11 – Google+ Pre-Test Configuration and Results

Google+											
Features	Tools used	Image Format Used	Cover Image File Name	Cover Image MD5	Stego Image File Name	Stego Image MD5	Downloaded Image File Name	Downloaded Image MD5	Successful Secret Message Extraction		Remarks
									Yes	No	
Photo Upload	JP Hide and Seek	JPEG	G_P1	EB6594636EE8ADC2E32400E2CEA7E7C6	SG_P1	A8A1EF7DD583A1615552DA8C6748F7C5	SG_P1.jpg	A8A1EF7DD583A1615552DA8C6748F7C5	x		Secret Message: secret.txt Secret Message Extraction: Successful
	Silent Eye	JPEG	G_P2	32B1C05D56EC4FE5A9CA268897C5287F	SG_P2	C87A274CF361E938132148DBABAD29AC	SG_P2.jpg	C87A274CF361E938132148DBABAD29AC	x		Luminance Interval = 5 JPG Quality = 30% Header position = bottom CharSet = ASCII Secret Message: Officials in many states said they have difficulties detecting forged birth certificates. Verifying date of birth is also required by the Act, and a system exists for doing so, but no licensing agencies are using it because of concerns about incomplete data, among other reasons. Partly because these two systems are not fully operational, GAO investigators were able to use counterfeit out-of-state drivers' licenses and birth certificates to fraudulently obtain licenses in three states. Secret Message Extraction: Successful
	EOF	JPEG	G_P3	9D958F2D4E34D6905372ECB17326B88F	SG_P3	07E412A1950D6D0C72D2156C0B676328	SG_P3.jpg	07E412A1950D6D0C72D2156C0B676328	x		Secret Message: The GAO's investigators obtained five driver's licenses in three different states under fictitious identities using combinations of names,

Google+											
Features	Tools used	Image Format Used	Cover Image File Name	Cover Image MD5	Stego Image File Name	Stego Image MD5	Downloaded Image File Name	Downloaded Image MD5	Successful Secret Message Extraction		Remarks
									Yes	No	
											<p>birthdates, and Social Security Numbers together with counterfeit documents. In two states, a GAO investigator was able to obtain two licenses with different identities using the same person's face. Only in one case did a motor vehicle employee appear to question the validity of the documents being presented—but the GAO investigator was still able to obtain a driver's license. "The investigators exploited cross state information vulnerabilities by using counterfeit driver's licenses from states other than the ones where they were issued valid driver's licenses. There's nothing to stop tomorrow's terrorists from obtaining valid driver's licenses in states where a pre-9/11 mentality prioritizes speedy customer service over the careful identity authentication of applicants. There are technological solutions to counter this risk, some of those made possible by federal systems and some from industry. The REAL ID Act identity security standards published in 2008 provide a framework for closing vulnerabilities like those exploited by GAO investigators, but compliance with those rules is voluntary," Zimmer noted.</p> <p>Secret Message Extraction: When the downloaded picture was opened with the Notepad application, the appended secret message is readable</p>

Features	Tools used	Image Format Used	Cover Image File Name	Cover Image MD5	Stego Image File Name	Stego Image MD5	Downloaded Image File Name	Downloaded Image MD5	Successful Secret Message Extraction		Remarks
									Yes	No	
	StegHid	BMP	G_P4	9FA076F0AC90AADC9C9595BAB2E74876	SG_P4	AC5C3D023707BA997336C484D6D007DC	SG_P4.bmp	AC5C3D023707BA997336C484D6D007DC	x		Encryption = Rijndael-128 Mode = cbc Compression = 9 Secret Message = secret.txt Secret Message Extraction: Successful
	S-Tools	GIF	G_P5	B2A9536AA6D5899B53D8370010C10B2B	SG_P5	621252150DDC2A68C62F81E7866B9635	SG_P5.gif	621252150DDC2A68C62F81E7866B9635	x		Encryption = IDEA Secret Message = secret.txt Secret Message Extraction: Successful
	Invisible Secret 4	PNG	G_P6	EAF67008E1F5980721C8A092621ED20E	SG_P6	8E57191A0B39542BA44A45FDE7DA50DA	SG_P6.png	8E57191A0B39542BA44A45FDE7DA50DA	x		Encryption = AES Rijndael Secret Message = secret.txt Secret Message Extraction: Successful

Appendix 12 – Scenario 1 Simulation Control Data (Target Machine 1 - Christian Riley)

Target machine 1 – Christian Riley				
Event	Date	Time	Action	Remarks
1	09 Dec 2012	15.14	Upload 2012-10-03_17-06-05_482.jpg with a message “download these pictures, you will love it” into Melody group with Facebook Add File feature	Cleaned picture MD5: 35BD3D93A9280B6C1521C28249AD35FC
2	09 Dec 2012	15.14 – 15.17	Created Steganographic image named 2012-10-20_19-20-03_927.jpg and saved it in folder named “special pictures”	
3	09 Dec 2012	15.17	Upload created 2012-10-20_19-20-03_927.jpg into Melody group with Facebook Add File feature using add file feature	Picture with secret message MD5: 559DCB0FDB6A1FA084F05F57EA66A181
4	09 Dec 2012	15.24	Upload 2012-10-22_15-43-29_300.jpg into Melody group with Facebook Add File feature using add file feature	Cleaned picture MD5: AB9FA0D6664314508DCEF9B4603B155B
8	09 Dec 2012	15.40 - 16.07	Performed Facebook chat with John Doe	
Facebook Chat as below:				
		15.40	hi christian i have downloaded it. What's nex?	John Doe
		15.42	great! Now go to this website: http://linux01.gwdg.de/~alatham/stego.html	Christin Riley
		15.43	download the window version	Christin Riley
		15.45	you need this software to get what you wanted	Christin Riley
		15.45	ok	John Doe
		15.48	ok got the software	John Doe
		15.49	do you think you know how to use it?	Christin Riley
		15.5	it's pretty simple	Christin Riley
		15.51	yes i guess so, but i think i need something to...?	John Doe
		15.53	yes. it is all in the file name, and i love numbers 4 from back	Christin Riley
		15.53	o..ok i think i got what you meant	John Doe
		15.54	i assume it is last four from left to right?	John Doe

Target machine 1 – Christian Riley				
Event	Date	Time	Action	Remarks
		15.55	yes	Christin Riley
		15.56	ok all unique	John Doe
		15.56	yup	Christin Riley
		15.56	great. give me a second, wanna try it out just to make sure we got this right	John Doe
		15.57	ok. if there is none to extract, it means none. just keep going until you got one	Christin Riley
		15.59	ok	John Doe
		16.02	ok I got it.	John Doe
		16.03	great! so same protocol in future and check for new post frequently in this melody group	Christin Riley
		16.03	ok	John Doe
		16.06	Oh one more thing..	Christin Riley
		16.07	Just hit on the Like once you have read the message..so that I know	Christin Riley
		16.07	ok	John Doe
9	09 Dec 2012	17.09	Download 2012-09-07_10-27-54_174.jpg from Melody group and save it in "from John" folder under "pictures" subfolder	
10	09 Dec 2012	17.1	Download 2012-09-21_21-13-51_504.jpg from Melody group and save it in "from John" folder under "pictures" subfolder	
11	09 Dec 2012	17.1	Download 2012-08-24_20-39-02_941.jpg from Melody group and save it in "from John" folder under "pictures" subfolder	
12	09 Dec 2012	17.10 - 17.12	Process secret message extraction with JPHS tool. Successful to extract a secret message from 2012-09-07_10-27-54_174.jpg and saved it as fromJohn.txt into "from John" folder under "pictures" subfolder	Secret message extracted
13	09 Dec 2012 - 10 Dec 2012	23.45 - 00.10	Created Steganographic image named 2012_12-01_16-42-35_679.jpg using command line and saved it in folder named "photos"	

Target machine 1 – Christian Riley				
Event	Date	Time	Action	Remarks
14	10 Dec 2012	00.11	Upload created 2012_12-01_16-42-35_679.jpg into Melody group with Facebook Add File feature using add file feature with a message "how about this?"	Picture with secret message MD5: 6B2129598C9BBA28AE6D905196BFB5AC
15	10 Dec 2012	00.11 - 00.15	Created Steganographic image named 2012-11-10_14-17-27_671.jpg using JPHS and saved it into folder named "special pictures"	
16	10 Dec 2012	00.16	Upload created 2012-11-10_14-17-27_671.jpg into Melody group with Facebook Add File feature using add file feature	Picture with secret message MD5: 6475B6592812C73E0515CCFEBF4B824E
17	10 Dec 2012	00.18	Upload 2012-12-01_17-46-00_497.jpg into Melody group with Facebook Add File feature using add file feature with a message "amazing ad.!"	Cleaned picture MD5: 2EDAE7EACD4324EF2AAA072DA4EB5C7E
18	10 Dec 2012	11.02	Download 2012-12-01_18-39-21_618.jpg from Melody group and save it in "from John" folder under "pictures" subfolder	
19	10 Dec 2012	11.02	Download 2012-10-05_15-22-29_347.jpg from Melody group and save it in "from John" folder under "pictures" subfolder	
20	10 Dec 2012	11.04	Download 2012-12-01_18-39-21_627.jpg from Melody group and save it in "from John" folder under "pictures" subfolder	
21	10 Dec 2012	11.05 - 11.06	Process secret message extraction with JPHS tool. Successful to extract a secret message from 2012-10-05_15-22-29_347.jpg and saved it as fromJohn1.txt into "from John" folder under "pictures" subfolder	Secret message extracted

**Appendix 13 – Scenario 1 Simulation Control Data
(Target Machine 2 - John Doe)**

Target machine 2 – John Doe				
Event	Date	Time	Action	Remarks
1	09 Dec 2012	15.33	Download 2012-10-22_15-43-29_300.jpg from Melody group and save it into "download" folder	
2	09 Dec 2012	16.00	Download 2012-10-03_17-06-05_482.jpg from Melody group and save it in "download" folder	
3	09 Dec 2012	16.00	Download 2012-10-20_19-20-03_927.jpg from Melody group and save it in "download" folder	
4	09 Dec 2012	16.01	Process secret message extraction with JPHS tool. Successful to extract a secret message from 2012-10-20_19-20-03_927.jpg and saved it as christ1.txt into "download" folder	Secret message extracted
4	09 Dec 2012	15.40 - 16.07	Performed Facebook chat with John Doe	
Facebook Chat as below:				
		15.40	hi christian i have downloaded it. What's nex?	John Doe
		15.42	great! Now go to this website: http://linux01.gwdg.de/~alatham/stego.html	Christin Riley
		15.43	download the window version	Christin Riley
		15.45	you need this software to get what you wanted	Christin Riley
		15.45	ok	John Doe
		15.48	ok got the software	John Doe
		15.49	do you think you know how to use it?	Christin Riley
		15.5	it's pretty simple	Christin Riley
		15.51	yes i guess so, but i think i need something to...?	John Doe
		15.53	yes. it is all in the file name, and i love numbers 4 from back	Christin Riley
		15.53	o..ok i think i got what you meant	John Doe
		15.54	i assume it is last four from left to right?	John Doe
		15.55	yes	Christin Riley

Target machine 2 – John Doe				
Event	Date	Time	Action	Remarks
		15.56	ok all unique	John Doe
		15.56	yup	Christin Riley
		15.56	great. give me a second, wanna try it out just to make sure we got this right	John Doe
		15.57	ok. if there is none to extract, it means none. just keep going until you got one	Christin Riley
		15.59	ok	John Doe
		16.02	ok I got it.	John Doe
		16.03	great! so same protocol in future and check for new post frequently in this melody group	Christin Riley
		16.03	ok	John Doe
		16.06	oh one more thing..	Christin Riley
		16.07	just hit on the Like once you have read the message..so that I know	Christin Riley
		16.07	ok	John Doe
5	09 Dec 2012	16.09	Upload 2012-08-24_20-39-02_941.jpg into Melody group with Facebook Add File feature	Cleaned picture MD5: 7FC5D2BA9D7C99A064F3E4F9257DAABC
6	09 Dec 2012	16.11 - 16.12	Created Steganographic image named 2012-09-07_10-27-54_174.jpg using JPHS and saved it into folder named “to Christ”	
7	09 Dec 2012	16.14	Upload created 2012-09-07_10-27-54_174.jpg into Melody group with Facebook Add File feature using add file feature.	Picture with secret message MD5: FED4AB0E6DE38C5EF938C7F4CCE3EDE7
8	09 Dec 2012	16.19	Upload 2012-09-21_21-13-51_504.jpg into Melody group with Facebook Add File feature	Cleaned picture MD5: BA9F748B1A33ACD186E4B10852F7AE77
9	09 Dec 2012	17.00	Deleted 2012-09-07_10-27-54_174.jpg from Melody group	deleted because secret message can't extracted from this download picture file
10	09 Dec 2012	17.02	Re-upload 2012-09-07_10-27-54_174.jpg into Melody group with Facebook Add File feature using add file feature.	Extraction performed on steganographic file that created previously and it works fine. So the same file was used for re-upload

Target machine 2 – John Doe				
Event	Date	Time	Action	Remarks
11	10 Dec 2012	00.34	Download 2012-12-01_17-46-00_497.jpg from Melody group and save it into "from Christ" folder under "pictures" subfolder	
12	10 Dec 2012	00.35	Download 2012_12-01_16-42-35_679.jpg from Melody group and save it into "from Christ" folder under "pictures" subfolder	
13	10 Dec 2012	00.35	Download 2012-11-10_14-17-27_671.jpg from Melody group and save it into "from Christ" folder under "pictures" subfolder	
14	10 Dec 2012	00.36	Moved 2012-10-03_17-06-05_482.jpg, 2012-10-20_19-20-03_927.jpg, 2012-10-22_15-43-29_300.jpg and christ1.txt from "download" folder to "from christ" folder under pictures subfolder	
15	10 Dec 2012	00.37	Process secret message extraction with JPHS tool. Successful to extract a secret message from 2012-11-10_14-17-27_671.jpg and save the extracted file as christ2.txt into "from christ" folder. The other file, 2012_12-01_16-42-35_679.jpg was successfully opened with the WinRAR program and extracted the embedded u.jpg file	Secret message extracted
16	10 Dec 2012	00.58	Upload 2012-12-01_18-39-21_618.jpg into Melody group with message "hohoho..happy holiday" using Facebook Add File feature	Cleaned picture MD5: B946809E2EB814D21 234C639356F3219
17	10 Dec 2012	00.58 - 01.00	Created Steganographic image named 2012-10-05_15-22-29_347.jpg using JPHS and saved it into folder named "to	

Target machine 2 – John Doe				
Event	Date	Time	Action	Remarks
			Christ”	
18	10 Dec 2012	01.00	Upload new created 2012-10-05_15-22-29_347.jpg into Melody group using Facebook Add File feature	Picture with secret message MD5: 3B89B5F316891B41E C4AAC619217ECDA
19	10 Dec 2012	01.01	Upload 2012-12-01_18-39-21_627.jpg into Melody group using Facebook Add File feature	Cleaned picture MD5: 4FCA948DF16C56930 E02B94F12BE1CEB

**Appendix 14 – Scenario 2 Simulation Control Data
(Target Machine 3 - John Doe)**


Target machine 3 – John Doe				
Event	Date	Time	Activities	Remarks
1	17 Nov 2012	7.36pm	created google+ circle named "benice"	
2	17 Nov 2012	7.36pm	Added Christian Riley into "benice" circle	
3	17 Nov 2012	7.39pm	posted message on "benice" circle "Hi christian, welcome to our club"	
4	17 Nov 2012	7.41pm	posted message on "benice" circle "Here are the treats that I love to share with you"	
5	17 Nov 2012	8.00pm	Executed StegHide.exe in usb flash drive, used cover image - IMG_2255 in usb flash drive, embedded with promo25_11.txt and saved the steganographic images as IMG_2255.jpg in picture folder with passphrase "2255". Encryption configured at default setting: Type = Rijndael-128; Mode = cbc	Steganographic Image: IMG_2255.jpg MD5: F4E533AD140BEFD2F05C58FB645E979A
6	17 Nov 2012	8.03pm	Executed StegHide.exe in usb flash drive, used cover image - IMG_7431 in usb flash drive, embedded with sales strategy.txt and saved the steganographic images as IMG_7431.jpg in picture folder with passphrase "2255". Encryption configured at default setting: Type = Rijndael-128; Mode = cbc	Steganographic Image: IMG_7431.jpg MD5: EBDEF169ED39CEA0BECAC80A6ACFE4F3
7	17 Nov 2012	8.05pm	Executed StegHide.exe in usb flash drive, used cover image - IMG_2488 in usb flash drive, embedded with market analysis.txt and saved the steganographic images as IMG_2488.jpg in picture folder with passphrase "2255". Encryption configured at default setting: Type = Rijndael-128; Mode = cbc	Steganographic Image: IMG_2488.jpg MD5: 74D4F86FE44F3B95D2E82FCBA6919559
8	17 Nov 2012	8.06pm	Executed StegHide.exe in usb flash drive, used cover image - IMG_6657 in usb flash drive, embedded with use of funds.txt and saved the steganographic images as IMG_6657.jpg in picture folder with passphrase	Steganographic Image: IMG_6657.jpg MD5: BA2B654BB65ACF79E444737C6C71C277

Target machine 3 – John Doe				
Event	Date	Time	Activities	Remarks
			"2255". Encryption configured at default setting: Type = Rijndael-128; Mode = cbc	
9	17 Nov 2012	8.09pm	upload IMG_2255, with message "Beautiful!"	Steganographic Image
10	17 Nov 2012	8.10pm	upload IMG_7431, with message "Had this for lunch yesterday, yummy!"	Steganographic Image
11	17 Nov 2012	8.10pm	upload IMG_6657, with message "on the way back home"	Steganographic Image
12	17 Nov 2012	8.11pm	upload IMG_2488, with message "you deserve this :)"	Steganographic Image
13	17 Nov 2012	8.11pm	Message posted on Buddy circle "hi christian, when are you free for a coffee?"	
14	17 Nov 2012	8.12pm	Message comment by Christian "how about this wed 6pm at Lone Café?"	
15	17 Nov 2012	8.13pm	Message comment by John Doe "ya sure! Will see you then"	
16	17 Nov 2012	8.13pm	Message comment by Christian "c u! remember to bring the tool to show me"	
17	17 Nov 2012	8.17pm	Message comment by John Doe "okie dokie"	
18	17 Nov 2012	8.28pm	Executed StegHide.exe in usb flash drive, used cover image - IMG_6677 in usb flash drive, embedded with promo02_12.txt and saved the steganographic images as IMG_6677.jpg in picture folder with passphrase "2255". Encryption configured at default setting: Type = Rijndael-128; Mode = cbc	Steganographic Image: IMG_6677.jpg MD5: 16598C670F034587 AC4A26C67D533B E7
19	17 Nov 2012	8.30pm	Executed StegHide.exe in usb flash drive, used cover image - IMG_8434 in usb flash drive, embedded with market analysis.txt and saved the steganographic images as IMG_8434.jpg in picture folder with passphrase "2255". Encryption configured at default setting: Type = Rijndael-128; Mode = cbc	Steganographic Image: IMG_8434.jpg MD5: 85B48065D865D8F6 F97B2CD46F15409 F

Target machine 3 – John Doe				
Event	Date	Time	Activities	Remarks
20	17 Nov 2012	8.33pm	Executed StegHide.exe in usb flash drive, used cover image - IMG_2292 in usb flash drive, embedded with cash flow projection.txt and saved the steganographic images as IMG_2292.jpg in picture folder with passphrase "2255". Encryption configured at default setting: Type = Rijndael-128; Mode = cbc	Steganographic Image: IMG_2292.jpg MD5: 8BC42CFBA965819 13D6CB1A96C9385 E2
21	17 Nov 2012	8.37pm	upload IMG_6677, with message "oh!"	Steganographic Image
22	17 Nov 2012	8.38pm	upload IMG_8434 with message "Shopping time!!"	Steganographic Image
23	17 Nov 2012	8.39pm	upload IMG_2292 with message "how about this.....^^"	Steganographic Image

Appendix 15 – Scenario 1 JP Hide and Seek’s artefacts detected by StegAlyzerAS (Target Machine 1 – Christian Riley)

Evidence Log: JPHide and JPSeek v0.51



Statistics

File Artifacts

Total File Artifacts Detected: 4/5

Percentage Detected: 80.0 %

False Positive/ Common File Artifacts Detected: 3/4

Unique File Artifacts Detected: 1/1

Registry Artifacts

Registry Artifacts Detected: 0

Percentage Detected: 0 %

Known File Artifacts

File	False Positive/Common	Status	Path	Time Stamp
jphide.exe	False Positive	**FOUND**	Detected Under: I:\Users\Christian\Documents\Christian\jphs_05\jphs05\jphide.exe	12/12/2012 2:04:50 p.m.
jphswin.exe	Unique	**FOUND**	Detected Under: I:\Users\Christian\Documents\Christian\jphs_05\jphs05\jphswin.exe	12/12/2012 2:04:50 p.m.
jpsseek.exe	False Positive	**FOUND**	Detected Under: I:\Users\Christian\Documents\Christian\jphs_05\jphs05\jpsseek.exe	12/12/2012 2:04:50 p.m.
Readme.txt	False Positive	**FOUND**	Multiple Detections (Double-Click for more information)	Multiple Detections (Double-Click for more information)
jphs_05.zip	False Positive	NOT FOUND	FILE NOT FOUND	FILE NOT FOUND

CRC32: 364C2731

MD5: 8E4D1C18E29DE5B5C071A59224D3236F

SHA1: BD8264C72CCAB4B981DA172A89F71B948E286637

SHA224: CF374B8DEB189B52B8DB2217A1A57707A47F0648BC445EBFD807B721

SHA256: A06ABA06E0785057101646A7155CFF24BF9716824C56B53F78D240F9B7C11C3B

SHA384: 40DDA576B62CDFD896486813C1D54DCA7D3FCCAA8278E97743B4238C0E2784676A4C3CD98EB7D3C8324E225218D7A22

SHA512: D10E692BE279437BCA24F352AF3C1C877B76BA9DA32343B658A00F650A008B613E96EF45CE948114988FC20CF465CB7FD42CB4488334CFD0B2D6A8256D

Size: 161280


Known Registry Artifacts

Message

There are currently no known registry artifacts associated with this application.

Appendix 16 – Scenario 1 JP Hide and Seek’s artefacts detected by StegAlyzerAS (Target Machine 2 – John Doe)

Evidence Log: JPHide and JPSeek v0.51



Statistics

File Artifacts

Total File Artifacts Detected: 5/5

Percentage Detected: 100 %

False Positive/ Common File Artifacts Detected: 4/4

Unique File Artifacts Detected: 1/1

Registry Artifacts

Registry Artifacts Detected: 0

Percentage Detected: 0 %

Known File Artifacts

File	False Positive/Common	Status	Path	Time Stamp
jphs_05.zip	False Positive	**FOUND**	Detected Under: R:\Users\John\Downloads\jphs_05.zip	12/12/2012 3:01:11 p.m.
jphide.exe	False Positive	**FOUND**	Detected Under: R:\Users\John\Downloads\jphs_05\jphs05\jphide.exe	12/12/2012 3:01:11 p.m.
jphswin.exe	Unique	**FOUND**	Detected Under: R:\Users\John\Downloads\jphs_05\jphs05\jphswin.exe	12/12/2012 3:01:11 p.m.
jpsseek.exe	False Positive	**FOUND**	Detected Under: R:\Users\John\Downloads\jphs_05\jphs05\jpsseek.exe	12/12/2012 3:01:11 p.m.
Readme.txt	False Positive	**FOUND**	Detected Under: R:\Users\John\Downloads\jphs_05\jphs05\Readme.txt	12/12/2012 3:01:11 p.m.

CRC32: C0CDB17D

MD5: DE067914621E4A61461259859624D211

SHA1: AD000FAA9D4A1C79D08384F8A8AD2EA1B68BF0E2

SHA224: B38E87697BD146BC9E65874A7BE816BD61E8F1395895824BD4E3C094

SHA256: 906E6080DAC971ACF27B42A1942702DAB27EF4E60DBB4DC61077DA429C5EEB67

SHA384: 86127910219C4C568832473E17A2136EEC1613DDDD5B4DFD0EFF07DFA19F0746BD2829839044280FBACADEBC2E32B69E

SHA512: 7C937EF1001B7006FB2C02226A6D8F57A022938448B5420AD0CE9C59E751472D500F18969D8EFCC455451095F669170A40086F4B4D27A489913E491BA9

Size: 184359


Known Registry Artifacts

Message

There are currently no known registry artifacts associated with this application.

Appendix 17 – Scenario 1 Bon Kyu Bon’s artefacts detected by StegAlyzerAS (Target Machines 1 & 2 - False Positive)

Evidence Log: Bon Kyu Bon v1.1.3011.2638



Statistics

File Artifacts

Total File Artifacts Detected: 2/12

Percentage Detected: 16.7 %

False Positive/ Common File Artifacts Detected: 0/1

Unique File Artifacts Detected: 2/11

Registry Artifacts

Registry Artifacts Detected: 0

Percentage Detected: 0 %

Known File Artifacts

File	False Positive/Common	Status	Path	Time Stamp
ilasm.exe.config	Unique	FOUND	Multiple Detections (Double-Click for more information)	Multiple Detections (Double-Click for more information)
ildasm.exe.config	Unique	FOUND	Multiple Detections (Double-Click for more information)	Multiple Detections (Double-Click for more information)
bkb.zip	Unique	NOT FOUND	FILE NOT FOUND	FILE NOT FOUND
Bon Kyu Bon.exe	Unique	NOT FOUND	FILE NOT FOUND	FILE NOT FOUND
Bon Kyu Bon.lnk	Unique	NOT FOUND	FILE NOT FOUND	FILE NOT FOUND
credits.txt	Unique	NOT FOUND	FILE NOT FOUND	FILE NOT FOUND
en-US.xml	Unique	NOT FOUND	FILE NOT FOUND	FILE NOT FOUND
fusion.dll	Unique	NOT FOUND	FILE NOT FOUND	FILE NOT FOUND

CRC32: E755468A

MD5: 0366F98E5EA426D80338070D8FA241B

SHA1: 153B90AF59D0598A0D5F5E083CB7FF24E2F7ADCF

SHA224: 11A9F71C4628EFC394628EBD0C1D89ACF79C38B886D41A23831B48AC

SHA256: 325B14941E79AE8570EB4062714D446F70B51D83C14FA58C5D2F90C8D0AFE3C3E

SHA384: 6D576A071AD9A038898138BD33399F7F15B1F38BA97C255F228F3CBF2824D4F99D63FBF964004FEA1F9E8AC076A6276F

SHA512: 563A39C5958AE6F507E37923959A8A2608C7E9A6F338053EDC142D8038849043C6050DF2946116876102704FF14D6B36314ACA468D91A7

Size: 181

Known Registry Artifacts

Message

There are currently no known registry artifacts associated with this application.

Locations at which ilasm.exe.config was detected for Bon Kyu Bon v1

File Name	Location
caspol.exe.config	R:\Windows\Microsoft.NET\Framework\v2.0.50727
ieexec.exe.config	R:\Windows\Microsoft.NET\Framework\v2.0.50727
ilasm.exe.config	R:\Windows\Microsoft.NET\Framework\v2.0.50727
regasm.exe.config	R:\Windows\Microsoft.NET\Framework\v2.0.50727
regsvcs.exe.config	R:\Windows\Microsoft.NET\Framework\v2.0.50727
caspol.exe.config	R:\Windows\winsxs\x86_caspol_b039f711d50a3a_6
ilasm.exe.config	R:\Windows\winsxs\x86_netfx-cir_ilasm_exe_b039f711d50a3a_6
ieexec.exe.config	R:\Windows\winsxs\x86_netfx-redist_config_files_b039f711d50a3a_6
regasm.exe.config	R:\Windows\winsxs\x86_regasm_b039f711d50a3a_6
regsvcs.exe.config	R:\Windows\winsxs\x86_regsvcs_b039f711d50a3a_6

**Appendix 18 – Scenario 1 Facebook File Download Artefacts
(Target Machine 1 – Christian Riley)**

Reconstructed File Download URLs History (Christian Riley's machine)			
Unix Time found in URL 2	Converted to Local Time	Facebook File Download URL 1	Facebook File Download URL 2
1355052147	Mon, 10 Dec 2012 00:22:27 +13:00	http://www.Facebook.com/download/286202834816056/2012_12-01_16-42-35_679.jpg	http://attachment.fbsbx.com/file_download.php?id=286202834816056&eid=ASuEUhcUSUyNqA_a33Qd6ap1WzZ-TGWYvSvfzcGuTcnkMsOGPleNr-gsnpcv8PItwJA&ext=1355052147&hash=ASuScI4oedm5GE-4
1355053264	Mon, 10 Dec 2012 00:41:04 +13:00	http://www.Facebook.com/download/387443488008206/2012-11-10_14-17-27_671.jpg	http://attachment.fbsbx.com/file_download.php?id=387443488008206&eid=ASu_EJXGoMGboXM7A9MWlhZezMt7xN goAVIBBtRfDfLA2-JZ5bxAOMLLsbSQJw6z2MA&ext=1355053264&hash=ASvEXKituArrkwp
1355090628	Mon, 10 Dec 2012 11:03:48 +13:00	http://www.Facebook.com/download/285914694845076/2012-10-05_15-22-29_347.jpg	http://attachment.fbsbx.com/file_download.php?id=285914694845076&eid=ASuUtOk791zisOwbi9Gu9iGZm_I4PuOo55qo4LHB1OSxf9qn6jcapUY62NU_BrvPWbC&ext=1355090628&hash=ASs5gD2dkwNW9Npa
1355090641	Mon, 10 Dec 2012 11:04:01 +13:00	http://www.Facebook.com/download/442744329123103/2012-12-01_18-39-21_618.jpg	http://attachment.fbsbx.com/file_download.php?id=442744329123103&eid=ASdywool9OfNxWBwrp-spnl_12f2_inowmTZ_1i5e4R6r_ZR8Mr426UowbLW-psmdm4&ext=1355090641&hash=ASsQ5pd-MYMBjWAw
1355090699	Mon, 10 Dec 2012 11:04:59 +13:00	http://www.Facebook.com/download/311763868932080/2012-12-01_18-39-21_627.jpg	http://attachment.fbsbx.com/file_download.php?id=311763868932080&eid=ASuaevSt0CqvIvJUrZpK5P40jI4NLgr7JPF GP-H2LPVFtNdpoyI8TerVmbcCHr1O20&ext=1355090699&ha

Reconstructed File Download URLs History (Christian Riley's machine)			
Unix Time found in URL 2	Converted to Local Time	Facebook File Download URL 1	Facebook File Download URL 2
			sh=ASsUSzWCyZx92KyE
1355020002	Sun, 9 Dec 2012 15:26:42 +13:00	http://www.Facebook.com/download/ 475114832531193 /2012-10-20_19-20-03_927.jpg	http://attachment.fbsbx.com/file_download.php?id= 475114832531193 &eid=ASuuik3jNhsCyOcZcjUHEYsWsglzEDV51bRQkRU009nQE0VTCC6Y_dGa63EDl5qwBkc&ext=1355020002&hash=ASvhrAdgvnPPABzK
1355026213	Sun, 9 Dec 2012 17:10:13 +13:00	http://www.Facebook.com/download/ 312232348881502 /2012-09-07_10-27-54_174.jpg	http://attachment.fbsbx.com/file_download.php?id= 312232348881502 &eid=AStwZUSuyYoV5Np-_1lnbHbNc9eqy8EJUWPhemBXpTxMa33vttD_d64Y_lUxfRHGNnl&ext=1355026213&hash=ASvsDVp2qLPoo3bk
1355026251	Sun, 9 Dec 2012 17:10:51 +13:00	http://www.Facebook.com/download/ 414901595250518 /2012-09-21_21-13-51_504.jpg	http://attachment.fbsbx.com/file_download.php?id= 414901595250518 &eid=ASsDf_Gm0Xo4Rr4TvT_svACjRNngsy8vCbR2bdf-R0VCIVYEGtuDA01iwociJu5OMls&ext=1355026251&hash=ASvqIPWB-EkRZtnO
1355026291	Sun, 9 Dec 2012 17:11:31 +13:00	http://www.Facebook.com/download/ 341523959279740 /2012-08-24_20-39-02_941.jpg	http://attachment.fbsbx.com/file_download.php?id= 341523959279740 &eid=ASuxR5Z_cXXIUZTujh-8Jpp_GLYQJgCBVn4jwRG8v5MDh14GYKLBP8AHllb0_OvoTTc&ext=1355026291&hash=ASv1g3OV2OO4IT1M

Notes:

Found the same file name in upload artefacts by **Christian Riley**

Found the same file name in upload artefacts by **John Doe**

**Appendix 19 – Scenario 1 Facebook File Download Artefacts
(Target Machine 2 – John Doe)**

Reconstructed File Download URLs History (John Doe's machine)			
Unix Time found in URL 2	Converted to Local Time	Facebook File Download URL 1	Facebook File Download URL 2
1355052889	Mon, 10 Dec 2012 00:34:49 +13:00	http://www.Facebook.com/download/411195238948694/2012-12-01_17-46-00_497.jpg	http://attachment.fbsbx.com/file_download.php?id=411195238948694&eid=ASsk7q44pCZWjH67lhcE9Ig5HV08nXpyRw8zoD4w1JD7Uau7In1zLStserDSNflXNpI&ext=1355052889&hash=AStz-V8hr75YmQC-
1355054606	Mon, 10 Dec 2012 01:03:26 +13:00	http://www.Facebook.com/download/285914694845076/2012-10-05_15-22-29_347.jpg	http://attachment.fbsbx.com/file_download.php?id=285914694845076&eid=ASsglnLbbNljfpk2B7z3iE-RkDOBFqgQ-4avt08XIOWcGkrlWY_qQKB6nPPjyKgMoXU&ext=1355054606&hash=ASsL8TvmERHrYpqD
1355055899	Mon, 10 Dec 2012 01:24:59 +13:00	http://www.Facebook.com/download/286202834816056/2012_12-01_16-42-35_679.jpg	http://attachment.fbsbx.com/file_download.php?id=286202834816056&eid=ASv7QWlt-geKOnGnGVRt4ADN8FsUPo8HhmLDIaGoWPKKy441r4TxAC0Jkazqbkt6FPg&ext=1355055899&hash=AStc-gjxtpsADWIL
1355056030	Mon, 10 Dec 2012 01:27:10 +13:00	http://www.Facebook.com/download/387443488008206/2012-11-10_14-17-27_671.jpg	http://attachment.fbsbx.com/file_download.php?id=387443488008206&eid=ASIdPbKAR3wR4OTH7CVKqRpRvINwLBBBrCA4xSM2f0_z5WnAE-wxzfqHZ6J2DDgRw&ext=1355056030&hash=ASsOQwZkwvqNvCrx
1355056079	Mon, 10 Dec 2012 01:27:59 +13:00	http://www.Facebook.com/download/475114832531193/2012-10-20_19-20-03_927.jpg	http://attachment.fbsbx.com/file_download.php?id=475114832531193&eid=AStymiFxfjvsTlXpm9-5EXt2p5VpJM2xOaC9eBSzJ22r2RWNjQUpQOwlnJpCvgkT5WYA&ext=1355056079&hash=ASuT9We205fjRsRQ

Reconstructed File Download URLs History (John Doe's machine)			
Unix Time found in URL 2	Converted to Local Time	Facebook File Download URL 1	Facebook File Download URL 2
1355056142	Mon, 10 Dec 2012 01:29:02 +13:00	http://www.Facebook.com/download/ 272228049566253 /2012-10-22_15-43-29_300.jpg	http://attachment.fbsbx.com/file_download.php?id= 272228049566253 &eid=ASstao7izc6oJ5RAuTnS9H-HPMNUb1NclqB4CYZ7mQWmMmO5Y5uhaX5iw2FOkiQghj0&ext=1355056142&hash=ASuoLE2a34KBWR1U
1355020383	Sun, 9 Dec 2012 15:33:03 +13:00	http://www.Facebook.com/download/ 476965689011882 /2012-10-03_17-06-05_482.jpg	http://attachment.fbsbx.com/file_download.php?id= 476965689011882 &eid=ASv2DRpbQCLv2isTZihy9KSC--6iucZsX3fZdss-7KrOQBVRNzvKsupO0hXWTHBPthg&ext=1355020383&hash=ASspD_gSddbG_Kue
1355023400	Sun, 9 Dec 2012 16:23:20 +13:00	http://www.Facebook.com/download/ 502514409778911 /2012-09-07_10-27-54_174.jpg	http://attachment.fbsbx.com/file_download.php?id= 502514409778911 &eid=AStzXS3lavOWWHG_Hu1YYr3YhznEvzRhuWMzg1rWncxoq9QzJurG66e408C5aWw9TVQ&ext=1355023400&hash=ASv_59WHAe0GUKHe
1355025863	Sun, 9 Dec 2012 17:04:23 +13:00	http://www.Facebook.com/download/ 312232348881502 /2012-09-07_10-27-54_174.jpg	http://attachment.fbsbx.com/file_download.php?id= 312232348881502 &eid=ASuU4KF173MNkGrpDmJp5zfPpVOSupkSWIQsqBY0bT49meXnNK4-fnv1DsAFvZrXNwI&ext=1355025863&hash=ASsuXuJ7kL0EztSc

Notes:

Found the same file name in upload artefacts by **Christian Riley**

Found the same file name in upload artefacts by **John Doe**

**Appendix 20 – Scenario 1 Facebook File Upload URL History
(Target Machine 1 – Christian Riley)**

No	File Type	MD5	Primary Device	True Path	Profile Name	Url Name	Internet Artifact Type	Record Last Accessed	Visit Count	URL Host
1	IE Cache Index dat	054a2324006d041cf976da6eb800ce90	Terrorism Related Case	Scenario1_Test2_CRiley\History\Visited Link\{d6cead4d-4241-11e2-a8c1-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752}	Christian	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=2&__iframe=true&__user=100003867343997	History\ Visited Link	09/12/12 03:14:03 p.m.	2	www.Facebook.com/
4	IE Cache Index dat	e50d57d8720a429c9b946389e9b9dd05	Terrorism Related Case	Scenario1_Test2_CRiley\History\Visited Link\{d6cead4d-4241-11e2-a8c1-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752}	Christian	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=3&__iframe=true&__user=100003867343997	History\ Visited Link	09/12/12 03:17:35 p.m.	2	www.Facebook.com/
5	IE Cache Index dat	bc8a6555d0da7e33842de123de0d3b12	Terrorism Related Case	Scenario1_Test2_CRiley\History\Visited Link\index.dat	Christian	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=4&__iframe=true&__user=100003867343997	History\ Visited Link	09/12/12 03:24:32 p.m.	2	www.Facebook.com/
6	IE Cache Index dat	bc8a6555d0da7e33842de123de0d3b12	Terrorism Related Case	Scenario1_Test2_CRiley\History\Visited Link\{d6cead4d-4241-11e2-a8c1-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752}	Christian	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=4&__iframe=true&__user=100003867343997	History\ Visited Link	09/12/12 03:24:32 p.m.	2	www.Facebook.com/

No	File Type	MD5	Primary Device	True Path	Profile Name	Url Name	Internet Artifact Type	Record Last Accessed	Visit Count	URL Host
10	IE Cache Index dat	bc8a6555d0da7e33842de123de0d3b12	Memory Dump	Scenario1_Test2_CRiley\History\Visited Link\Unallocated Clusters	Christian	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=4&__iframe=true&__user=100003867343997	History\ Visited Link	09/12/12 03:24:32 p.m.	2	www.Facebook.com/
2	IE Cache Index dat	25e2455434aa249bf131c482843a14b6	Terrorism Related Case	Scenario1_Test2_CRiley\History\Visited Link\index.dat	Christian	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=2&__iframe=true&__user=100003867343997	History\ Visited Link	10/12/12 12:11:13 a.m.	4	www.Facebook.com/
7	IE Cache Index dat	25e2455434aa249bf131c482843a14b6	Memory Dump	Scenario1_Test2_CRiley\History\Visited Link\Unallocated Clusters	Christian	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=2&__iframe=true&__user=100003867343997	History\ Visited Link	10/12/12 12:11:13 a.m.	4	www.Facebook.com/
3	IE Cache Index dat	18a28a619f42d34cfa606f1b8138dc9a	Terrorism Related Case	Scenario1_Test2_CRiley\History\Visited Link\index.dat	Christian	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=3&__iframe=true&__user=100003867343997	History\ Visited Link	10/12/12 12:16:28 a.m.	4	www.Facebook.com/
8	IE Cache Index dat	18a28a619f42d34cfa606f1b8138dc9a	Memory Dump	Scenario1_Test2_CRiley\History\Visited Link\Unallocated Clusters	Christian	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=3&__iframe=true&__user=100003867343997	History\ Visited Link	10/12/12 12:16:28 a.m.	4	www.Facebook.com/
9	IE Cache Index dat	d72fa643c1007d40f07d99d0bc9031b8	Memory Dump	Scenario1_Test2_CRiley\History\Visited Link\Unallocated Clusters	Christian	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=4&__iframe=true&__user=100003867343997	History\ Visited Link	10/12/12 12:18:31 a.m.	4	www.Facebook.com/

**Appendix 21 – Scenario 1 Facebook File Upload URL History
(Target Machine 2 – John Doe)**

No	File Type	MD5	Primary Device	True Path	Profile Name	URL Name	Internet Artifact Type	Record Last Accessed	Visit Count	URL Host
5	IE Cache Index dat	6c8096af839e a5f6f47c1d19 b15eabb9	Terrorism Related Case	Scenario1_Test2_JDoe \\History\\Visited Link\\index.dat	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=3&__iframe=true&__user=100003861284061	History\\Visited Link	09/12/12 04:09:57 p.m.	2	www.Facebook.com/
6	IE Cache Index dat	6c8096af839e a5f6f47c1d19 b15eabb9	Memory Dump	Scenario1_Test2_JDoe \\History\\Visited Link\\Unallocated Clusters	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=3&__iframe=true&__user=100003861284061	History\\Visited Link	09/12/12 04:09:57 p.m.	2	www.Facebook.com/
7	IE Cache Index dat	6af2ba03bbf9 935f96bfd794 c213e7e1	Terrorism Related Case	Scenario1_Test2_JDoe \\History\\Visited Link\\index.dat	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=4&__iframe=true&__user=100003861284061	History\\Visited Link	09/12/12 04:12:52 p.m.	2	www.Facebook.com/
8	IE Cache Index dat	6af2ba03bbf9 935f96bfd794 c213e7e1	Memory Dump	Scenario1_Test2_JDoe \\History\\Visited Link\\Unallocated Clusters	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=4&__iframe=true&__user=100003861284061	History\\Visited Link	09/12/12 04:12:52 p.m.	2	www.Facebook.com/
9	IE Cache Index dat	31d3165ff1bf8 b4db88713b09 e71ebad	Terrorism Related Case	Scenario1_Test2_JDoe \\History\\Visited Link\\index.dat	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=5&__iframe=true&__user=100003861284061	History\\Visited Link	09/12/12 04:19:30 p.m.	2	www.Facebook.com/

No	File Type	MD5	Primary Device	True Path	Profile Name	URL Name	Internet Artifact Type	Record Last Accessed	Visit Count	URL Host
10	IE Cache Index dat	31d3165ff1bf8b4db88713b09e71ebad	Memory Dump	Scenario1_Test2_JDoe\History\Visited Link\Unallocated Clusters	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=5&__iframe=true&__user=100003861284061	History\ Visited Link	09/12/12 04:19:30 p.m.	2	www.Facebook.com/
11	IE Cache Index dat	f9a9ccf0cc6eb53fbc216795ba079828	Terrorism Related Case	Scenario1_Test2_JDoe\History\Visited Link\index.dat	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=2&__iframe=true&__user=100003861284061	History\ Visited Link	09/12/12 05:02:49 p.m.	2	www.Facebook.com/
12	IE Cache Index dat	f9a9ccf0cc6eb53fbc216795ba079828	Memory Dump	Scenario1_Test2_JDoe\History\Visited Link\Unallocated Clusters	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=2&__iframe=true&__user=100003861284061	History\ Visited Link	09/12/12 05:02:49 p.m.	2	www.Facebook.com/
13	IE Cache Index dat	4acd2b769f6338c84a4ce4378f622ea3	Terrorism Related Case	Scenario1_Test2_JDoe\History\Visited Link\index.dat	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=9&__iframe=true&__user=100003861284061	History\ Visited Link	10/12/12 01:00:19 a.m.	2	www.Facebook.com/
14	IE Cache Index dat	4acd2b769f6338c84a4ce4378f622ea3	Memory Dump	Scenario1_Test2_JDoe\History\Visited Link\Unallocated Clusters	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=9&__iframe=true&__user=100003861284061	History\ Visited Link	10/12/12 01:00:19 a.m.	2	www.Facebook.com/
1	IE Cache Index dat	4dbca6926bbe61be5022d609bca03bf3	Terrorism Related Case	Scenario1_Test2_JDoe\History\Visited Link\index.dat	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=10&__iframe=true&__user=100003861284061	History\ Visited Link	10/12/12 01:01:11 a.m.	2	www.Facebook.com/

No	File Type	MD5	Primary Device	True Path	Profile Name	URL Name	Internet Artifact Type	Record Last Accessed	Visit Count	URL Host
2	IE Cache Index dat	4dbca6926bbe61be5022d609bca03bf3	Memory Dump	Scenario1_Test2_JDoe\History\Visited Link\Unallocated Clusters	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=10&__iframe=true&__user=100003861284061	History\Visited Link	10/12/12 01:01:11 a.m.	2	www.Facebook.com/
3	IE Cache Index dat	3147967c2a7ed0c1d3a88189bab7064b	Terrorism Related Case	Scenario1_Test2_JDoe\History\Visited Link\index.dat	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=8&__iframe=true&__user=100003861284061	History\Visited Link	10/12/12 12:58:22 a.m.	2	www.Facebook.com/
4	IE Cache Index dat	3147967c2a7ed0c1d3a88189bab7064b	Memory Dump	Scenario1_Test2_JDoe\History\Visited Link\Unallocated Clusters	John	http://www.Facebook.com/ajax/groups/files/upload?__a=1&__adt=8&__iframe=true&__user=100003861284061	History\Visited Link	10/12/12 12:58:22 a.m.	2	www.Facebook.com/

**Appendix 22 – Scenario 1 Facebook File Upload Artefacts
(Target Machine 1 – Christian Riley)**

Time	File upload artefacts in Facebook	Found in
Sun, 9 Dec 2012 15:17:32 +13:00	<p>·i·d·=1·0·0·0·0·3·8·6·7·3·4·3·9·9·7·\·"·>·C·h·r·i·s·t·i·a·n· ·R·i·l·e·y·<·/·a·>· ·u·p·l·o·a·d·e·d· ·a· ·f·i·l·e· /·2·0·1·2·--·1·0·--·2·0·_·1·9·--·2·0·--·0·3·_·9·2·7·\·.·j·p·g·\·". ·r·e·l·=·\·"·i·g·n·o·r·e·\·"·>·D·o·w·n·l·o·a·d·<·/·a·> ·/·a·j·a·x·/·g·r·o·u·p·s·/·f·i·l·e·s·/·r·e·v·i·s·i·o·n·?·m·e·s·s·a·g·e·_·i·d·=1·7·9·9·8·7·9·0·2·1·4·5·5 ·7·6·\·"· ·r·o·l·e·=·\·"·b·u·t·t·o·n·\·"·>·U·p·l·o·a·d· ·R·e·v·i·s·i·o·n·<·/·a·> ·c·o·n·t·e·n·t·_·t·i·m·e·s·t·a·m·p·&·q·u·o·t·;·:·&·q·u·o·t·;·1·3·5·5·0·1·9·4·5·2·&·q·u·o·t·</p>	Scenario1_Test2_CRiley\Terrorism Related Case\D\pagefile.sys
Sun, 9 Dec 2012 15:24:29 +13:00	<p>·i·d·=1·0·0·0·0·3·8·6·7·3·4·3·9·9·7·"·>·C·h·r·i·s·t·i·a·n· ·R·i·l·e·y·<·/·a·>· ·u·p·l·o·a·d·e·d· ·a· ·f·i·l·e· <·s·p·a·n· ·c·l·a·s·s·="·f·w·b· ·f·c·b·"·>·2·0·1·2·--·1·0·--·2·2·_·1·5·--·4·3·-- ·2·9·_·3·0·0·.·j·p·g·<·/·s·p·a·n·> ·/·a·j·a·x·/·g·r·o·u·p·s·/·f·i·l·e·s·/·r·e·v·i·s·i·o·n·?·m·e·s·s·a·g·e·_·i·d·=1·7·9·9·8·9·1·5·2·1·4·5·4 ·5·1·"· ·r·o·l·e·="·b·u·t·t·o·n·"·>·U·p·l·o·a·d· ·R·e·v·i·s·i·o·n·<·/·a·> ·d·a·t·a·-·u·t·i·m·e·="·1·3·5·5·0·1·9·8·6·9·"· ·c·l·a·s·s·="·t·i·m·e·s·t·a·m·p·</p>	Scenario1_Test2_CRiley\Terrorism Related Case\D\pagefile.sys
Sun, 9 Dec 2012 15:13:59 +13:00	<p>·u·s·e·r·.·p·h·p·?·i·d·=1·0·0·0·0·3·8·6·7·3·4·3·9·9·7·"·>·C·h·r·i·s·t·i·a·n· ·R·i·l·e·y·<·/·a·>· ·u·p·l·o·a·d·e·d· ·a· ·f·i·l·e·.·<·/·h·5·> <·s·p·a·n· ·c·l·a·s·s·="·f·w·b· ·f·c·b·"·>·2·0·1·2·--·1·0·--·0·3·_·1·7·--·0·6·-- ·0·5·_·4·8·2·.·j·p·g·<·/·s·p·a·n·></p>	Scenario1_Test2_CRiley\Terrorism Related Case\D\pagefile.sys

Time	File upload artefacts in Facebook	Found in
	<p>/a.j.a.x./m.e.s.s.a.g.i.n.g./a.t.t.a.c.h.m.e.n.t.s./p.h.o.t.o./d.i.a.l.o.g..p.h.p.?u.r.i.=.%2F.d.o.w.n.l.o.a.d.%2F4.7.6.9.6.5.6.8.9.0.1.1.8.8.2.%2F2.0.1.2.--1.0--0.3_.1.7--0.6--0.5_.4.8.2..j.p.g".</p> <p>.a. .c.l.a.s.s="u.i.L.i.n.k.L.i.g.h.t.B.l.u.e".</p> <p>.h.r.e.f="."/d.o.w.n.l.o.a.d./4.7.6.9.6.5.6.8.9.0.1.1.8.8.2./2.0.1.2.--1.0--0.3_.1.7--0.6--0.5_.4.8.2..j.p.g". .r.e.l="i.g.n.o.r.e">D.o.w.n.l.o.a.d</p> <p>a.j.a.x.i.f.y="."/a.j.a.x./g.r.o.u.p.s./f.i.l.e.s./r.e.v.i.s.i.o.n.?m.e.s.s.a.g.e._i.d=1.7.9.9.8.7.2.8.8.8.1.2.3.0.4". .r.o.l.e="b.u.t.t.o.n">U.p.l.o.a.d. R.e.v.i.s.i.o.n.</p> <p>.d.a.t.a--u.t.i.m.e="1.3.5.5.0.1.9.2.3.9". .c.l.a.s.s="t.i.m.e.s.t.a.m.p.</p>	
Mon, 10 Dec 2012 00:18:27 +13:00	<p>/user.php?id=100003867343997">Christian Riley uploaded a file.</div></h5></p> <p>amazing ad.!!^</p> <p><i class="_8o _8r lfloat img sp_czc6sg sx_266747"></i><div class="_8m _8u">2012-12-01_17-46-00_497.jpg</p> <p>href="#" rel="dialog"</p> <p>ajaxify="/ajax/messaging/attachments/photo/dialog.php?uri=%2Fdownload%2F411195238948694%2F2012-12-01_17-46-00_497.jpg" role="button">Preview</p> <p>href="/download/411195238948694/2012-12-01_17-46-00_497.jpg" rel="ignore">Download</p> <p>/ajax/groups/files/revision?message_id=180120852132281" role="button">Upload Revision</p>	<p>Scenario1_Test2_CRiley\Terrorism Related Case\D\pagefile.sys</p> <p>Scenario1_Test2_CRiley\Terrorism Related Case\D\Blocks.mem</p> <p>Scenario1_Test2_CRiley\Terrorism Related Case\D\nacl_irt_x86_32.nexe</p>

Time	File upload artefacts in Facebook	Found in
	"content_timestamp":"1355051907" data-utime="1355051907" class="timestamp"	
Mon, 10 Dec 2012 00:16:24 +13:00	/user.php?id=100003867343997">Christian Riley uploaded a file.</h5> <i class="_8o _8r lfloat img sp_czc6sg sx_266747"></i><div class="_8m _8u">2012-11-10_14-17-27_671.jpg ajaxify="/ajax/messaging/attachments/photo/dialog.php?uri=%2Fdownload%2F387443488008206%2F2012-11-10_14-17-27_671.jpg" role="button">Preview href="/download/387443488008206/2012-11-10_14-17-27_671.jpg" rel="ignore">Download Upload Revision content_timestamp":"1355051784"	Scenario1_Test2_CRiley\Terrorism Related Case\D\pagefile.sys Scenario1_Test2_CRiley\Terrorism Related Case\D\nacl_irt_x86_32.nexe Scenario1_Test2_CRiley\Terrorism Related Case\D\Blocks.mem
Sun, 9 Dec 2012 15:13:59 +13:00	/user.php?id=100003867343997">Christian Riley uploaded a file.</div><div class="messageBody">download these pictures, you will love it! <i class="_8o _8r lfloat img sp_czc6sg sx_266747"></i><div class="_8m _8u">2012-10-03_17-06-05_482.jpg	Scenario1_Test2_CRiley\Terrorism Related Case\D\pagefile.sys

Time	File upload artefacts in Facebook	Found in
	<pre> ajaxify="/ajax/messaging/attachments/photo/dialog.php?uri=%2Fdownload%2F476965689011882%2F2012-10-03_17-06-05_482.jpg" role="button">Preview href="/download/476965689011882/2012-10-03_17-06-05_482.jpg" rel="ignore">Download ajaxify="/ajax/groups/files/revision?message_id=179987288812304" role="button">Upload Revision< "content_timestamp":"1355019239" data-utime="1355019239" class="timestamp </pre>	
Sun, 9 Dec 2012 15:24:29 +13:00	<pre> /user.php?id=100003867343997">Christian Riley uploaded a file.</h5> <i class="_8o _8r lfloat img sp_czc6sg sx_266747"></i><div class="_8m _8u">2012-10-22_15-43-29_300.jpg ajaxify="/ajax/messaging/attachments/photo/dialog.php?uri=%2Fdownload%2F272228049566253%2F2012-10-22_15-43-29_300.jpg" role="button">Preview href="/download/272228049566253/2012-10-22_15-43-29_300.jpg" rel="ignore">Download Upload Revision ,"content_timestamp":"1355019869" Sunday, 9 December 2012 at 15:24 data-utime="1355019869" class="timestamp </pre>	<p>Scenario1_Test2_CRiley\Terrorism Related Case\D\pagefile.sys</p> <p>Scenario1_Test2_CRiley\Terrorism Related Case\D\nacl_irt_x86_32.nexe</p> <p>Scenario1_Test2_CRiley\Terrorism Related Case\D\Blocks.mem</p>

Time	File upload artefacts in Facebook	Found in
Sun, 9 Dec 2012 15:17:32 +13:00	<p>/user.php?id=100003867343997">Christian Riley uploaded a file.</h5></p> <p><i class="_8o _8r lfloat img sp_czc6sg sx_266747"></i><div class="_8m _8u">2012-10-20_19-20-03_927.jpg</p> <p>Preview</p> <p>Download</p> <p>Upload Revision</p> <p>content_timestamp"&quot;;&quot;1355019452&quot;</p>	<p>Scenario1_Test2_CRiley\Terrorism Related Case\D\pagefile.sys</p> <p>Scenario1_Test2_CRiley\Terrorism Related Case\D\nacl_irt_x86_32.nexe</p> <p>Scenario1_Test2_CRiley\Terrorism Related Case\D\Blocks.mem</p>
Mon, 10 Dec 2012 00:11:09 +13:00	<p>/user.php?id=100003867343997\">Christian Riley\u003C/a> uploaded a file.</p> <p>span class=\"messageBody\">\u003Cspan class=\"userContent\">how about this?</p> <p>Ci class=\"_8o _8r lfloat img sp_czc6sg sx_266747\">\u003C/i>\u003Cdiv class=\"_8m _8u\">\u003Cspan class=\"fwb fcb\">2012_12-01_16-42-35_679.jpg</p> <p>ajaxify=\"\ajax/messaging/attachments/photo/dialog.php?uri=\u00252Fdownload\u00252F286202834816056\u00252F2012_12-01_16-42-35_679.jpg\" role=\"button\">Preview</p>	<p>Scenario1_Test2_CRiley\Terrorism Related Case\D\Blocks.mem</p> <p>Scenario1_Test2_CRiley\Terrorism Related Case\D\nacl_irt_x86_32.nexe</p>

Time	File upload artefacts in Facebook	Found in
	uiLinkLightBlue\" href=\"\\download\\286202834816056\\2012_12-01_16-42-35_679.jpg\" rel=\"ignore\">Download\\ /ajax\\groups\\files\\revision?message_id=180119332132433\" role=\"button\">Upload Revision\\ content_timestamp";"1355051469" "Monday, 10 December 2012 at 00:11\" data-utime=\"1355051469\"	
Sun, 9 Dec 2012 16:09:56 +13:00	/.u.s.e.r..p.h.p.?i.d.=1.0.0.0.0.3.8.6.1.2.8.4.0.6.1.\">H.a.p.p.y. F.a.r.m.<./a.>. .u.p.l.o.a.d.e.d. a .f.i.l.e..<./h.5> <d.i.v. c.l.a.s.s=\"_8.m. _8.u.\"><s.p.a.n. c.l.a.s.s=\"f.w.b. f.c.b.\">2.0.1.2---0.8.- .2.4._2.0---3.9---0.2._9.4.1..j.p.g.<./s.p.a.n.> .a.s.s.o.c._o.b.j._i.d.&q.u.o.t;.:&q.u.o.t;1.7.2.8.8.8.1.6.9.5.2.2.1.6.&q.u.o.t;. .c.o.n.t.e.n.t._t.i.m.e.s.t.a.m.p.&q.u.o.t;.:&q.u.o.t;1.3.5.5.0.2.2.5.9.6.&q.u.o.t.	Scenario1_Test2_CRiley\\Terrorism Related Case\\D\\pagefile.sys
Sun, 9 Dec 2012 16:19:29 +13:00	/user.php?id=100003861284061\">Happy Farm uploaded a file.</h5> <i class=\"_8o _8r lfloat img sp_czc6sg sx_266747\"></i><div class=\"_8m _8u\">2012-09-21_21-13-51_504.jpg ;assoc_obj_id";"172888169522216" content_timestamp";"1355023169"	Scenario1_Test2_CRiley\\Terrorism Related Case\\D\\pagefile.sys Scenario1_Test2_CRiley\\Terrorism Related Case\\D\\Blocks.mem Scenario1_Test2_CRiley\\Terrorism Related Case\\D\\nacl_irt_x86_32.nexe

Time	File upload artefacts in Facebook	Found in
Mon, 10 Dec 2012 00:58:20 +13:00	\user.php?id=100003861284061\">Happy Farm\<a> uploaded a file.\</div>\ \<div class=\"_8m _8u\">\2012-12-01_18-39-21_618.jpg\\ "172888169522216" content_timestamp":"1355054300";	Scenario1_Test2_CRiley\Terrorism Related Case\D\TorchTorrent.exe Scenario1_Test2_CRiley\Terrorism Related Case\D\Blocks.mem Scenario1_Test2_CRiley\Terrorism Related Case\D\nacl_irt_x86_32.nexe Scenario1_Test2_CRiley\Terrorism Related Case\D\Unallocated Clusters
Sun, 9 Dec 2012 17:02:48 +13:00	/user.php?id=100003861284061\">Happy Farm\<a> uploaded a file.\</h5> >\<div class=\"_8m _8u\">\2012-09-07_10-27-54_174.jpg\ assoc_obj_id":"172888169522216" ;content_timestamp":"1355025768";	Scenario1_Test2_CRiley\Terrorism Related Case\D\Blocks.mem Scenario1_Test2_CRiley\Terrorism Related Case\D\{d6cead4d-4241-11e2-a8c1-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752} Scenario1_Test2_CRiley\Terrorism Related Case\D\nacl_irt_x86_32.nexe
Sun, 9 Dec 2012 16:09:56 +13:00	/user.php?id=100003861284061\">Happy Farm\<a> uploaded a file.\</h5> \<div class=\"_8m _8u\">\2012-08-24_20-39-02_941.jpg\\ 	Scenario1_Test2_CRiley\Terrorism Related Case\D\Blocks.mem Scenario1_Test2_CRiley\Terrorism Related Case\D\nacl_irt_x86_32.nexe

Time	File upload artefacts in Facebook	Found in
	assoc_obj_id";"172888169522216" content_timestamp";"1355022596"	Scenario1_Test2_CRiley\Terrorism Related Case\D\Unallocated Clusters

**Appendix 23 – Scenario 1 Facebook File Upload Artefacts
(Target Machine 2 – John Doe)**

Time	File upload artefacts in Facebook	Found in
Sun, 9 Dec 2012 16:19:29 +13:00	<p>/user.php?id=100003861284061">Happy Farm\u003C/a> uploaded a file.\</p> <p>3Ci class=\"_8o _8r lfloat img sp_czc6sg sx_266747\">\u003C/i>\u003Cdiv class=\"_8m _8u\">\u003Cspan class=\"fbw fcb\">2012-09-21_21-13-51_504.jpg</p> <p>ajaxify=\"\ajax/messaging/attachments/photo/dialog.php?uri=\u00252Fdownload\u00252F414901595250518\u00252F2012-09-21_21-13-51_504.jpg\</p> <p>href=\"\download/414901595250518/2012-09-21_21-13-51_504.jpg\"</p> <p>rel=\"ignore\">Download</p> <p>ajaxify=\"\ajax/groups/files/revision?message_id=180002148810818\"</p> <p>role=\"button\">Upload Revision</p> <p>;assoc_obj_id&quot;;&quot;172888169522216&quot</p> <p>;content_timestamp&quot;;&quot;1355023169</p>	Scenario1_Test2_JDoe\Terrorism Related Case\D\pagefile.sys
Sun, 9 Dec 2012 16:19:29 +13:00	<p>/a.j.a.x/.h.o.v.e.r.c.a.r.d/.u.s.e.r..p.h.p?.i.d.=1.0.0.0.0.3.8.6.1.2.8.4.0.6.1.\">.H. a.p.p.y. .F.a.r.m.<./a>. .u.p.l.o.a.d.e.d. .a. .f.i.l.e..<./h.5. <s.p.a.n. .c.l.a.s.s.=\"f.w.b. .f.c.b.\">.2.0.1.2.--0.9.--2.1._.2.1.--1.3.-- .5.1._.5.0.4..j.p.g.<./s.p.a.n>. h.r.e.f.=\"./d.o.w.n.l.o.a.d./4.1.4.9.0.1.5.9.5.2.5.0.5.1.8./2.0.1.2.--0.9.--2.1._.2.1.-- .1.3.--5.1._.5.0.4..j.p.g.\". .r.e.l.=\".i.g.n.o.r.e.\">.D.o.w.n.l.o.a.d.<./a> ./a.j.a.x/.g.r.o.u.p.s/.f.i.l.e.s/.r.e.v.i.s.i.o.n?.m.e.s.s.a.g.e._i.d.=1.8.0.0.0.2.1.4. 8.8.1.0.8.1.8.\". .r.o.l.e.=\".b.u.t.t.o.n.\">.U.p.l.o.a.d. .R.e.v.i.s.i.o.n .a.s.s.o.c._o.b.j._i.d.&.q.u.o.t;.:&.q.u.o.t;.:1.7.2.8.8.8.1.6.9.5.2.2.2.1.6.</p>	Scenario1_Test2_JDoe\Terrorism Related Case\D\pagefile.sys

Time	File upload artefacts in Facebook	Found in
	c·o·n·t·e·n·t_·t·i·m·e·s·t·a·m·p·&·q·u·o·t;·;·&·q·u·o·t;;1·3·5·5·0·2·3·1·6·9	
Sun, 9 Dec 2012 16:12:51 +13:00	i·d·=1·0·0·0·0·3·8·6·1·2·8·4·0·6·1·">·H·a·p·p·y· ·F·a·r·m·<·/·a·>· ·u·p·l·o·a·d·e·d· ·a· ·f·i·l·e· ·<·d·i·v· ·c·l·a·s·s·="·_·8·m· ·_·8·u·">·<·s·p·a·n· ·c·l·a·s·s·="·f·w·b· ·f·c·b·">·2·0·1·2·--0·9·--0·7·_·1·0·--2·7·--5·4·_·1·7·4·.·j·p·g /·a·j·a·x·/·m·e·s·s·a·g·i·n·g·/·a·t·t·a·c·h·m·e·n·t·s·/·p·h·o·t·o·/·d·i·a·l·o·g·.·p·h·p·?·u·r·i· =·%·2·F·d·o·w·n·l·o·a·d·%·2·F·5·0·2·5·1·4·4·0·9·7·7·8·9·1·1·%·2·F·2·0·1·2·--0·9·-- ·0·7·_·1·0·--2·7·--5·4·_·1·7·4·.·j·p·g· ·h·r·e·f·="·/·d·o·w·n·l·o·a·d·/·5·0·2·5·1·4·4·0·9·7·7·8·9·1·1·/·2·0·1·2·--0·9·-- ·0·7·_·1·0·--2·7·--5·4·_·1·7·4·.·j·p·g·"· ·r·e·l·="·i·g·n·o·r·e·">·D·o·w·n·l·o·a·d· /·a·j·a·x·/·g·r·o·u·p·s·/·f·i·l·e·s·/·r·e·v·i·s·i·o·n·?·m·e·s·s·a·g·e·_·i·d·=1·8·0·0·0·0·9·8·2 ·1·4·4·2·6·8·"· ·r·o·l·e·="·b·u·t·t·o·n·">·U·p·l·o·a·d· ·R·e·v·i·s·i·o·n ·a·s·s·o·c·_·o·b·j·_·i·d·&·q·u·o·t;·;·&·q·u·o·t;;1·7·2·8·8·8·1·6·9·5·2·2·2·1·6·&·q·u·o·t ·;· c·o·n·t·e·n·t_·t·i·m·e·s·t·a·m·p·&·q·u·o·t;·;·&·q·u·o·t;;1·3·5·5·0·2·2·7·7·1	Scenario1_Test2_JDoe\Terrorism Related Case\D\pagefile.sys
Sun, 9 Dec 2012 16:09:56 +13:00	·u·s·e·r·.·p·h·p·?·i·d·=1·0·0·0·0·3·8·6·1·2·8·4·0·6·1·">·H·a·p·p·y· ·F·a·r·m·<·/·a·>· ·u·p·l·o·a·d·e·d· ·a· ·f·i·l·e·.·< ·<·d·i·v· ·c·l·a·s·s·="·_·8·m· ·_·8·u·">·<·s·p·a·n· ·c·l·a·s·s·="·f·w·b· ·f·c·b·">·2·0·1·2·--0·8·--2·4·_·2·0·--3·9·--0·2·_·9·4·1·.·j·p·g· /·a·j·a·x·/·m·e·s·s·a·g·i·n·g·/·a·t·t·a·c·h·m·e·n·t·s·/·p·h·o·t·o·/·d·i·a·l·o·g·.·p·h·p·?·u·r·i· =·%·2·F·d·o·w·n·l·o·a·d·%·2·F·3·4·1·5·2·3·9·5·9·2·7·9·7·4·0·%·2·F·2·0·1·2·--0·8·-- ·2·4·_·2·0·--3·9·--0·2·_·9·4·1·.·j·p·g·	Scenario1_Test2_JDoe\Terrorism Related Case\D\pagefile.sys

Time	File upload artefacts in Facebook	Found in
	<p>·h·r·e·f·=·"/·d·o·w·n·l·o·a·d·/·3·4·1·5·2·3·9·5·9·2·7·9·7·4·0·/·2·0·1·2·--0·8·-· ·2·4·_·2·0·--3·9·--0·2·_·9·4·1·.·j·p·g·"· ·r·e·l·=·"·i·g·n·o·r·e·">·D·o·w·n·l·o·a·d· /·a·j·a·x·/·g·r·o·u·p·s·/·f·i·l·e·s·/·r·e·v·i·s·i·o·n·?·m·e·s·s·a·g·e·_·i·d·=·1·8·0·0·0·0·4·4·5· ·4·7·7·6·5·5·"· ·r·o·l·e·=·"·b·u·t·t·o·n·">·U·p·l·o·a·d· ·R·e·v·i·s·i·o·n ·a·s·s·o·c·_·o·b·j·_·i·d·&·q·u·o·t·;·:·&·q·u·o·t·;·1·7·2·8·8·8·1·6·9·5·2·2·2·1·6 c·o·n·t·e·n·t·_·t·i·m·e·s·t·a·m·p·&·q·u·o·t·;·:·&·q·u·o·t·;·1·3·5·5·0·2·2·5·9·6</p>	
Mon, 10 Dec 2012 01:01:09 +13:00	<p>user.php?id=100003861284061\">Happy Farm\u003C/a> uploaded a file. class=\"_8o _8r lfloat img sp_czc6sg sx_266747\">\u003C/i>\u003Cdiv class=\"_8m _8u\">\u003Cspan class=\"fwb fcb\">2012-12-01_18-39-21_627.jpg /ajax/messaging/attachments/photo/dialog.php?uri=\u00252Fdownload\u00252F31176 3868932080\u00252F2012-12-01_18-39-21_627.jpg\" role=\"button\">Preview href=\"\"/download/311763868932080/2012-12-01_18-39-21_627.jpg\" rel=\"ignore\">Download ajaxify=\"\"/ajax/groups/files/revision?message_id=180131758797857\" role=\"button\">Upload Revision\ assoc_obj_id&quot;;&quot;;172888169522216 content_timestamp&quot;;&quot;;1355054469</p>	Terrorism Related Case\D\Unallocated Clusters
Sun, 9 Dec 2012 15:24:29 +13:00	<p>/USER.PHP?ID=100003867343997\">CHRISTIAN RILEY\u003C/A>UPLOADED A FILE IN THE GROUP\u003CA CLASS=\"PRONOUN- LINK\"HREF=\"\"/GROUPS/172888169522216\"/\"DATA- FT=\"&#123;&QUOT;TN&QUOT;;&QUOT;A&QUOT;&#125;\">MELODY\u003C/A</p>	Scenario1_Test2_JDoe\Terrorism Related Case\D\pagefile.sys

Time	File upload artefacts in Facebook	Found in
	<p>></p> <p>\U003CSPAN CLASS=\ "FWB FCB\ ">2012-10-22_15-43-29_300.JPG\U003C\SPAN> &QUOT;172888169522216&QUOT; ;CONTENT_TIMESTAMP&QUOT;;&QUOT;1355019869&QUOT;</p>	
Sun, 9 Dec 2012 15:17:32 +13:00	<p>/USER.PHP?ID=100003867343997\ ">CHRISTIAN RILEY\U003C\A>UPLOADED A FILE IN THE GROUP\U003CA CLASS=\ "PRONOUN- LINK\ "HREF=\ "\GROUPS\172888169522216\ "\DATA- FT=\ "&#123;&QUOT;TN&QUOT;;&QUOT;A&QUOT;&#125;\ ">MELODY\U003C\A > U003CSPAN CLASS=\ "FWB FCB\ ">2012-10-20_19-20-03_927.JPG\U003C\SPAN> ASSOC_OBJ_ID&QUOT;;&QUOT;172888169522216&QUOT; ;CONTENT_TIMESTAMP&QUOT;;&QUOT;1355019452&QUOT;</p>	Scenario1_Test2_JDoe\Terrorism Related Case\D\pagefile.sys
Sun, 9 Dec 2012 15:17:32 +13:00	<p>/·u·s·e·r·.·p·h·p·?·i·d·=·1·0·0·0·0·3·8·6·7·3·4·3·9·9·7·...>·C·h·r·i·s·t·i·a·n· ·R·i·l·e·y·<·/·a·>· ·u·p·l·o·a·d·e·d· ·a· ·f·i·l·e·.·<·/·d·i·v·>·<·/·h·5·>· <·s·p·a·n· ·c·l·a·s·s·... "·m·e·s·s·a·g·e·B·o·d·y·...>·<·s·p·a·n· ·c·l·a·s·s·... "·u·s·e·r·C·o·n·t·e·n·t·...>·d·o·w·n·l·o·a·d· ·t·h·e·s·e· ·p·i·c·t·u·r·e·s·,· ·y·o·u· ·w·i·l·l· ·l·o·v·e· ·i·t·!<·/·s·p·a·n·>· <·i· ·c·l·a·s·s·... "·_·8·o· ·_·8·r· ·l·f·l·o·a·t· ·i·m·g· ·s·p·_·c·z·c·6·s·g· ·s·x·_·2·6·6·7·4·7·...>·<·/·i·>·<·d·i·v· ·c·l·a·s·s·... "·_·8·m· ·_·8·u·...>·<·s·p·a·n· ·c·l·a·s·s·... "·f·w·b· ·f·c·b·...>·2·0·1·2·--·1·0·--·0·3·_·1·7·--·0·6·- ·0·5·_·4·8·2·.·j·p·g·<·/·s·p·a·n·>· <·a· ·h·r·e·f·... "/·g·r·o·u·p·s·/·1·7·2·8·8·8·1·6·9·5·2·2·2·1·6·/·1·7·9·9·8·7·8·9·8·8·1·2·2·4·3·</p>	Scenario1_Test2_JDoe\Terrorism Related Case\D\pagefile.sys

[illegible]

Time	File upload artefacts in Facebook	Found in
Sun, 9 Dec 2012 15:13:59 +13:00	<p>/user.php?id=100003867343997">Christian Riley\u003C/a> uploaded a file in the group \u003Ca class=\"pronoun-link \" href=\"\"/groups/172888169522216\" data- ft=\"&#123;&quot;tn&quot;;&quot;A&quot;&#125;\">Melody\u003C/a>.\u003C/div>\u 003C/h5></p> <p>\u003Cspan class=\"fbw fcb\">2012-10-03_17-06-05_482.jpg\u003C/span></p> <p>assoc_obj_id&quot;;&quot;172888169522216&quot; ;content_timestamp&quot;;&quot;1355019239&quot;</p>	Scenario1_Test2_JDoe\Terrorism Related Case\D\Unallocated Clusters

Appendix 24 – Scenario 1 Images of Interest in Suspects’ Hard Drives

Suspect's Name	Name	Comment	File Ext	Category	Last Accessed	MD5	Item Path	Identified as Steganographic Images	Secret Message Extracted
Christian Riley	2012-12-01 17-46-00_497.jpg		jpg	Picture	09/12/12 01:05:43p.m.	2edae7eacd4324ef2aaa072da4eb5c7e	Terrorism Related Case\D\Users\Christian\Picture s\Photos\2012-12-01_17-46-00_497.jpg	negative	
Christian Riley	2012-10-03 17-06-05_482.jpg		jpg	Picture	09/12/12 01:05:43p.m.	35bd3d93a9280b6c1521c28249ad35fc	Terrorism Related Case\D\Users\Christian\Picture s\Photos\2012-10-03_17-06-05_482.jpg	negative	
Christian Riley	2012-10-05 15-22-29_347.jpg	same file name as FB download URL	jpg	Picture	10/12/12 11:02:40a.m.	3b89b5f316891b41ec4aac619217ecda	Terrorism Related Case\D\Users\Christian\Picture s\from John\2012-10-05_15-22-29_347.jpg	yes (*)	yes
Christian Riley	2012-12-01 18-39-21_627.jpg	same file name as FB download URL	jpg	Picture	10/12/12 11:04:12a.m.	4fca948df16c56930e02b94f12be1ceb	Terrorism Related Case\D\Users\Christian\Picture s\from John\2012-12-01_18-39-21_627.jpg	negative	
Christian Riley	2012-10-20 19-20-03_927.jpg	same file name as FB download URL	jpg	Picture	09/12/12 03:25:59p.m.	559dcb0fdb6a1fa084f05f57ea66a181	Terrorism Related Case\D\Users\Christian\Downl oads\2012-10-20_19-20-03_927.jpg	yes (***)	Yes
Christian Riley	2012-10-20 19-20-03_927.jpg	same file name as FB download URL	jpg	Picture	09/12/12 03:16:51p.m.	559dcb0fdb6a1fa084f05f57ea66a181	Terrorism Related Case\D\Users\Christian\Picture s\Special pictures\2012-10-20_19-20-03_927.jpg	yes (***)	Yes
Christian Riley	2012-11-10 14-17-27_671.jpg	same file name as FB download URL	jpg	Picture	10/12/12 12:22:20a.m.	6475b6592812c73e0515ccfebf4b	Terrorism Related Case\D\Users\Christian\Downl oads\2012-11-10_14-17-	yes (*)	Yes

Suspect's Name	Name	Comment	File Ext	Category	Last Accessed	MD5	Item Path	Identified as Steganographic Images	Secret Message Extracted
						824e	27_671.jpg		
Christian Riley	2012-11-10 14-17-27_671.jpg	same file name as FB download URL	jpg	Picture	10/12/12 12:14:37a.m.	6475b6592812c73e0515ccfebf4b824e	Terrorism Related Case\D\Users\Christian\Pictures\Special pictures\2012-11-10_14-17-27_671.jpg	yes (*)	Yes
Christian Riley	2012_12-01_16-42-35_679.jpg	same file name as FB download URL	jpg	Picture	10/12/12 12:21:41a.m.	6b2129598c9bba28ae6d905196bf b5ac	Terrorism Related Case\D\Users\Christian\Downloads\2012_12-01_16-42-35_679.jpg	yes (appended)	Yes
Christian Riley	2012_12-01_16-42-35_679.jpg	same file name as FB download URL	jpg	Picture	09/12/12 11:57:25p.m.	6b2129598c9bba28ae6d905196bf b5ac	Terrorism Related Case\D\Users\Christian\Pictures\Photos\2012_12-01_16-42-35_679.jpg	yes (appended)	Yes
Christian Riley	2012-10-20_19-20-03_927.jpg	Same file name as FB download URL, different MD5	jpg	Picture	09/12/12 01:05:43p.m.	6f7991608d9f13591ff70073ac453580	Terrorism Related Case\D\Users\Christian\Pictures\Photos\2012-10-20_19-20-03_927.jpg	negative	
Christian Riley	2012-12-01_16-42-35_678.jpg	different file name & MD5, but same picture display as 2012_12-01_16-42-35_679.jpg	jpg	Picture	09/12/12 01:05:43p.m.	7f01238d832f0470d819483e72628802	Terrorism Related Case\D\Users\Christian\Pictures\Photos\2012-12-01_16-42-35_678.jpg		
Christian Riley	2012-08-24_20-39-02_941.jpg	same file name as FB download URL	jpg	Picture	09/12/12 05:10:41p.m.	7fc5d2ba9d7c99a064f3e4f9257daa bc	Terrorism Related Case\D\Users\Christian\Pictures\from John\2012-08-24_20-39-02_941.jpg	negative	

Suspect's Name	Name	Comment	File Ext	Category	Last Accessed	MD5	Item Path	Identified as Steganographic Images	Secret Message Extracted
Christian Riley	2012-10-22_15-43-29_300.jpg		jpg	Picture	09/12/12 01:05:43p.m.	ab9fa0d6664314508dc ef9b4603b155b	Terrorism Related Case\D\Users\Christian\Picture s\Photos\2012-10-22_15-43-29_300.jpg	negative	
Christian Riley	2012-12-01_18-39-21_618.jpg	same file name as FB download URL	jpg	Picture	10/12/12 11:02:21a.m.	b946809e2eb814d212 34c639356f3219	Terrorism Related Case\D\Users\Christian\Picture s\from John\2012-12-01_18-39-21_618.jpg	negative	
Christian Riley	2012-09-21_21-13-51_504.jpg	same file name as FB download URL	jpg	Picture	09/12/12 05:10:21p.m.	ba9f748b1a33acd186e 4b10852f7ae77	Terrorism Related Case\D\Users\Christian\Picture s\from John\2012-09-21_21-13-51_504.jpg	negative	
Christian Riley	2012-11-10_14-17-27_671.jpg	same name as FB download URL, same file name, different MD5	jpg	Picture	09/12/12 01:05:43p.m.	c27525ca5a91ba58dc4 c52799a4bca48	Terrorism Related Case\D\Users\Christian\Picture s\Photos\2012-11-10_14-17-27_671.jpg	negative	
Christian Riley	2012-09-07_10-27-54_174.jpg	same name as FB download URL	jpg	Picture	09/12/12 05:09:43p.m.	fed4ab0e6de38c5ef938 c7f4cce3ede7	Terrorism Related Case\D\Users\Christian\Picture s\from John\2012-09-07_10-27-54_174.jpg	yes (***)	Yes
John Doe	2012-12-01_17-46-00_497.jpg	same name as FB download URL	jpg	Picture	10/12/12 12:34:22a.m.	2edae7eacd4324ef2aaa 072da4eb5c7e	Terrorism Related Case\D\Users\John\Pictures\fro m Christ\2012-12-01_17-46-00_497.jpg	negative	
John Doe	2012-10-03_17-06-05_482.jpg	same name as FB download URL	jpg	Picture	10/12/12 12:36:36a.m.	35bd3d93a9280b6c15 21c28249ad35fc	Terrorism Related Case\D\Users\John\Pictures\fro m Christ\2012-10-03_17-06-05_482.jpg	negative	

Suspect's Name	Name	Comment	File Ext	Category	Last Accessed	MD5	Item Path	Identified as Steganographic Images	Secret Message Extracted
John Doe	2012-10-05_15-22-29_347.jpg	same name as FB download URL	jpg	Picture	10/12/12 01:03:10a.m.	3b89b5f316891b41ec4aac619217ecda	Terrorism Related Case\D\Users\John\Downloads\special photos\2012-10-05_15-22-29_347.jpg	yes (*)	yes
John Doe	2012-10-05_15-22-29_347.jpg	same name as FB download URL	jpg	Picture	10/12/12 12:56:52a.m.	3b89b5f316891b41ec4aac619217ecda	Terrorism Related Case\D\Users\John\Pictures\To Christ\2012-10-05_15-22-29_347.jpg	yes (*)	Yes
John Doe	2012-12-01_18-39-21_627.jpg	same name as FB download URL	jpg	Picture	09/12/12 01:58:35p.m.	4fca948df16c56930e02b94f12be1ceb	Terrorism Related Case\D\Users\John\Pictures\Photos\2012-12-01_18-39-21_627.jpg	negative	
John Doe	2012-10-20_19-20-03_927.jpg	same file name as FB download URL	jpg	Picture	10/12/12 12:36:36a.m.	559dcb0fdb6a1fa084f05f57ea66a181	Terrorism Related Case\D\Users\John\Pictures\from Christ\2012-10-20_19-20-03_927.jpg	Yes (***)	Yes
John Doe	2012-09-07_10-27-54_174.jpg	Same file name as FB download URL, different MD5	jpg	Picture	09/12/12 01:58:35p.m.	5ac2497d7a3359070dcea457a658e436	Terrorism Related Case\D\Users\John\Pictures\Photos\2012-09-07_10-27-54_174.jpg	yes (*) false positive	No
John Doe	2012-11-10_14-17-27_671.jpg	same file name as FB download URL	jpg	Picture	10/12/12 12:35:34a.m.	6475b6592812c73e0515ccfebf4b824e	Terrorism Related Case\D\Users\John\Pictures\from Christ\2012-11-10_14-17-27_671.jpg	yes (*)	Yes
John Doe	2012_12-01_16-42-35_679.jpg	same file name as FB download URL	jpg	Picture	10/12/12 12:35:03a.m.	6b2129598c9bba28ae6d905196bf b5ac	Terrorism Related Case\D\Users\John\Pictures\from Christ\2012_12-01_16-42-35_679.jpg	yes (appended)	Yes

Suspect's Name	Name	Comment	File Ext	Category	Last Accessed	MD5	Item Path	Identified as Steganographic Images	Secret Message Extracted
John Doe	2012-08-24 20-39-02_941.jpg		jpg	Picture	09/12/12 01:58:35p.m.	7fc5d2ba9d7c99a064f3e4f9257daabc	Terrorism Related Case\D\Users\John\Pictures\Photos\2012-08-24_20-39-02_941.jpg	negative	
John Doe	2012-10-22 15-43-29_300.jpg	same file name as FB download URL	jpg	Picture	10/12/12 12:36:36a.m.	ab9fa0d6664314508dc ef9b4603b155b	Terrorism Related Case\D\Users\John\Pictures\from Christ\2012-10-22_15-43-29_300.jpg	negative	
John Doe	2012-12-01 18-39-21_618.jpg		jpg	Picture	09/12/12 01:58:35p.m.	b946809e2eb814d21234c639356f3219	Terrorism Related Case\D\Users\John\Pictures\Photos\2012-12-01_18-39-21_618.jpg	negative	
John Doe	2012-09-21 21-13-51_504.jpg		jpg	Picture	09/12/12 01:58:35p.m.	ba9f748b1a33acd186e4b10852f7ae77	Terrorism Related Case\D\Users\John\Pictures\Photos\2012-09-21_21-13-51_504.jpg	negative	
John Doe	2012-10-05 15-22-29_347.jpg	same file name as FB download URL, different MD5	jpg	Picture	09/12/12 01:58:35p.m.	e7471bfa75eb9136fd37b88e0a91b211	Terrorism Related Case\D\Users\John\Pictures\Photos\2012-10-05_15-22-29_347.jpg	negative	
John Doe	2012-09-07 10-27-54_174.jpg	same file name as FB download URL	jpg	Picture	09/12/12 04:12:22p.m.	fed4ab0e6de38c5ef938c7f4cce3ede7	Terrorism Related Case\D\Users\John\Pictures\To Christ\2012-09-07_10-27-54_174.jpg	yes (***)	Yes
John Doe	2012-09-07 10-27-54_174.jpg	same file name as FB download URL	jpg	Picture	09/12/12 05:03:34p.m.	fed4ab0e6de38c5ef938c7f4cce3ede7	Terrorism Related Case\D\Users\John\Downloads\special photos\2012-09-07_10-27-54_174.jpg	yes (***)	Yes

Appendix 25 – Scenario 1 Facebook Chat Artefacts from pagefile.sys and unallocated cluster

```
"a.u.t.h.o.r.": ".f.b.i.d.:1.0.0.0.0.3.8.6.1.2.8.4.0.6.1.", "a.u.t.h.o.r._e.m.a.i.l.":
".1.0.0.0.0.3.8.6.1.2.8.4.0.6.1.\u0040f.a.c.e.b.o.o.k..c.o.m.", "c.o.o.r.d.i.n.a
.t.e.s.": ".n.u.l.l.", "t.i.m.e.s.t.a.m.p.": "1.3.5.5.0.2.0.8.2.3.0.7.2.", "t.i.m.e.s.t.a.m.p
._a.b.s.o.l.u.t.e.": ".T.o.d.a.y.", "t.i.m.e.s.t.a.m.p._r.e.l.a.t.i.v.e.": ".1.5.:4.0.",
".i.s._u.n.r.e.a.d.": ".f.a.l.s.e.", ".i.s._f.o.r.w.a.r.d.": ".f.a.l.s.e.", ".i.s._f.i.l.t.e.r.e.d
._c.o.n.t.e.n.t.": ".f.a.l.s.e.", ".f.o.r.w.a.r.d._c.o.u.n.t.": "0.", ".f.o.r.w.a.r.d._m.e.s.s
.a.g.e._i.d.s.": ".n.u.l.l.", ".s.o.u.r.c.e.": ".s.o.u.r.c.e.:c.h.a.t.:w.e.b.", ".s.o.u.r.c.e
._t.a.g.s.": "[.s.o.u.r.c.e.:c.h.a.t.].", ".s.p.o.o.f._w.a.r.n.i.n.g.": ".f.a.l.s.e.", ".f.o.l
.d.e.r.": ".i.n.b.o.x.", ".b.o.d.y.": ".h.i.c.h.r.i.s.t.i.a.n.i.h.a.v.e.d.o.w.n.l.o.a
.d.e.d.i.t..W.h.a.t.'s.n.e.x.?", ".h.t.m.l._b.o.d.y.": ".n.u.l.l.", ".s.u.b.j.e.c.t.": ".n
.u.l.l.", ".h.a.s._a.t.t.a.c.h.m.e.n.t.": ".f.a.l.s.e.", ".a.t.t.a.c.h.m.e.n.t.s.": "[.]", ".r.a.w
._a.t.t.a.c.h.m.e.n.t.s.": ".n.u.l.l.", ".t.h.r.e.a.d._i.d.": ".i.d.:2.4.2.2.3.7.0.4.2.5.4.8.
1.7.6.", ".a.c.t.i.o.n._i.d.": ".1.3.5.5.0.2.0.8.2.4.0.8.1.0.0.0.0.0.", ".a.c.t.i.o.n
._t.y.p.e.": ".m.a--t.y.p.e.:u.s.e.r--g.e.n.e.r.a.t.e.d-
.m.e.s.s.a.g.e."}, {"m.e.s.s.a.g.e._i.d.": ".m.s.g.:3.d.4.5.f.7.4.9.c.a.3.f.b.5.b.9.
2.f.d.3.4.d.c.f.f.9.5.e.8.c.5.1.4.4.", ".t.h.r.e.a.d.i.n.g._i.d.": ".\u003C.1.3.5.5.
0.2.0.9.6.1.4.2.9.:2.8.6.4.8.1.0.9.8.2-
.2.8.9.0.8.0.7.0.8.5.\u0040m.a.i.l.p.r.o.j.e.k.t.i.t.a.n..c.o.m.>.",
```

```
"a.u.t.h.o.r.": ".f.b.i.d.:1.0.0.0.0.3.8.6.7.3.4.3.9.9.7.", "a.u.t.h.o.r._e.m.a.i.l.":
".1.0.0.0.0.3.8.6.7.3.4.3.9.9.7.\u0040f.a.c.e.b.o.o.k..c.o.m.", "c.o.o.r.d.i.n.a
.t.e.s.": ".n.u.l.l.", "t.i.m.e.s.t.a.m.p.": "1.3.5.5.0.2.0.9.5.8.8.2.7.", "t.i.m.e.s.t.a.m.p
._a.b.s.o.l.u.t.e.": ".T.o.d.a.y.", "t.i.m.e.s.t.a.m.p._r.e.l.a.t.i.v.e.": ".1.5.:4.2.",
".i.s._u.n.r.e.a.d.": ".f.a.l.s.e.", ".i.s._f.o.r.w.a.r.d.": ".f.a.l.s.e.", ".i.s._f.i.l.t.e.r.e.d
._c.o.n.t.e.n.t.": ".f.a.l.s.e.", ".f.o.r.w.a.r.d._c.o.u.n.t.": "0.", ".f.o.r.w.a.r.d._m.e.s.s
.a.g.e._i.d.s.": ".n.u.l.l.", ".s.o.u.r.c.e.": ".s.o.u.r.c.e.:c.h.a.t.:w.e.b.", ".s.o.u.r.c.e
._t.a.g.s.": "[.s.o.u.r.c.e.:c.h.a.t.].", ".s.p.o.o.f._w.a.r.n.i.n.g.": ".f.a.l.s.e.", ".f.o.l
.d.e.r.": ".i.n.b.o.x.", ".b.o.d.y.": ".g.r.e.a.t!.N.o.w.g.o.y.t.o.t.h.i.s.w.e.b.s.i
.t.e.:h.t.t.p.://.l.i.n.u.x.0.1.g.w.d.g.d.e./~a.l.a.t.h.a.m./s.t.e.g.o.h.t.m
.l.", ".h.t.m.l._b.o.d.y.": ".n.u.l.l.", ".s.u.b.j.e.c.t.": ".n.u.l.l.", ".h.a.s._a.t.t.a.c.h.m.e
.n.t.": ".f.a.l.s.e.", ".a.t.t.a.c.h.m.e.n.t.s.": "[.]", ".r.a.w._a.t.t.a.c.h.m.e.n.t.s.": ".n.u.l.l
.", ".t.h.r.e.a.d._i.d.": ".i.d.:2.4.2.2.3.7.0.4.2.5.4.8.1.7.6.", ".a.c.t.i.o.n._i.d.": ".
1.3.5.5.0.2.0.9.5.8.9.9.2.0.0.0.0.0.", ".a.c.t.i.o.n._t.y.p.e.": ".m.a-
.t.y.p.e.:u.s.e.r--g.e.n.e.r.a.t.e.d-
.m.e.s.s.a.g.e."}, {"m.e.s.s.a.g.e._i.d.": ".m.s.g.:a.8.c.4.4.5.d.3.1.f.3.1.e.a.0.9.
1.6.a.5.d.2.5.c.e.b.c.a.4.b.7.7.0.4.", ".t.h.r.e.a.d.i.n.g._i.d.": ".\u003C.1.3.5.5.
0.2.0.9.8.7.0.9.1.:4.4.4.1.6.8.7.1.9-
.2.8.9.0.8.0.7.0.8.5.\u0040m.a.i.l.p.r.o.j.e.k.t.i.t.a.n..c.o.m.>.",
```

```
"a.u.t.h.o.r.": ".f.b.i.d.:1.0.0.0.0.3.8.6.7.3.4.3.9.9.7.", "a.u.t.h.o.r._e.m.a.i.l.":
".1.0.0.0.0.3.8.6.7.3.4.3.9.9.7.\u0040f.a.c.e.b.o.o.k..c.o.m.", "c.o.o.r.d.i.n.a
```

·t·e·s·":·n·u·l·l·,"t·i·m·e·s·t·a·m·p·":·1·3·5·5·0·2·0·9·8·4·4·7·3·,"t·i·m·e·s·t·a·m·p·
_a·b·s·o·l·u·t·e·":·"T·o·d·a·y·","t·i·m·e·s·t·a·m·p·_r·e·l·a·t·i·v·e·":·"1·5·:·4·3·",
·"i·s·_u·n·r·e·a·d·":·f·a·l·s·e·,"i·s·_f·o·r·w·a·r·d·":·f·a·l·s·e·,"i·s·_f·i·l·t·e·r·e·d·
_c·o·n·t·e·n·t·":·f·a·l·s·e·,"f·o·r·w·a·r·d·_c·o·u·n·t·":·0·,"f·o·r·w·a·r·d·_m·e·s·s·
a·g·e·_i·d·s·":·n·u·l·l·,"s·o·u·r·c·e·":·"s·o·u·r·c·e·:·c·h·a·t·:·w·e·b·","s·o·u·r·c·e·
_t·a·g·s·":·["s·o·u·r·c·e·:·c·h·a·t·"],,"s·p·o·o·f·_w·a·r·n·i·n·g·":·f·a·l·s·e·,"f·o·l·
·d·e·r·":·"i·n·b·o·x·","b·o·d·y·":·"d·o·w·n·l·o·a·d·_t·h·e·_w·i·n·d·o·w·_v·e·r·s·i·o
·n·","h·t·m·l·_b·o·d·y·":·n·u·l·l·,"s·u·b·j·e·c·t·":·n·u·l·l·,"h·a·s·_a·t·t·a·c·h·m·e
·n·t·":·f·a·l·s·e·,"a·t·t·a·c·h·m·e·n·t·s·":·[],,"r·a·w·_a·t·t·a·c·h·m·e·n·t·s·":·n·u·l·
l·,"t·h·r·e·a·d·_i·d·":·"i·d·.2·4·2·2·3·7·0·4·2·5·4·8·1·7·6·","a·c·t·i·o·n·_i·d·":·"
·1·3·5·5·0·2·0·9·8·4·7·4·5·0·0·0·0·0·0·","a·c·t·i·o·n·_t·y·p·e·":·"m·a--
·t·y·p·e·:·u·s·e·r--g·e·n·e·r·a·t·e·d--
·m·e·s·s·a·g·e·"}·,{·"m·e·s·s·a·g·e·_i·d·":·"m·s·g·.8·9·5·f·6·2·4·d·7·a·b·3·8·1·f·b·
3·c·5·9·7·0·d·d·e·e·0·7·b·3·c·1·9·9·","t·h·r·e·a·d·i·n·g·_i·d·":·"\u003C·1·3·5·5
·0·2·1·1·1·7·1·7·4·:·3·1·2·1·0·2·7·9·4·6·-
·2·3·1·7·9·1·8·6·0·0·\u0040·m·a·i·l·.p·r·o·j·e·k·t·i·t·a·n·.c·o·m>"}·,

"a·u·t·h·o·r·":·"f·b·i·d·:·1·0·0·0·0·3·8·6·7·3·4·3·9·9·7·","a·u·t·h·o·r·_e·m·a·i·l·":·
·"1·0·0·0·0·3·8·6·7·3·4·3·9·9·7·\u0040·f·a·c·e·b·o·o·k·.c·o·m",,"c·o·o·r·d·i·n·a
·t·e·s·":·n·u·l·l·,"t·i·m·e·s·t·a·m·p·":·1·3·5·5·0·2·1·1·1·4·5·6·0·,"t·i·m·e·s·t·a·m·p·
_a·b·s·o·l·u·t·e·":·"T·o·d·a·y·","t·i·m·e·s·t·a·m·p·_r·e·l·a·t·i·v·e·":·"1·5·:·4·5·",
·"i·s·_u·n·r·e·a·d·":·f·a·l·s·e·,"i·s·_f·o·r·w·a·r·d·":·f·a·l·s·e·,"i·s·_f·i·l·t·e·r·e·d·
_c·o·n·t·e·n·t·":·f·a·l·s·e·,"f·o·r·w·a·r·d·_c·o·u·n·t·":·0·,"f·o·r·w·a·r·d·_m·e·s·s·
a·g·e·_i·d·s·":·n·u·l·l·,"s·o·u·r·c·e·":·"s·o·u·r·c·e·:·c·h·a·t·:·w·e·b·","s·o·u·r·c·e·
_t·a·g·s·":·["s·o·u·r·c·e·:·c·h·a·t·"],,"s·p·o·o·f·_w·a·r·n·i·n·g·":·f·a·l·s·e·,"f·o·l·
·d·e·r·":·"i·n·b·o·x·","b·o·d·y·":·"y·o·u·_n·e·e·d·_t·h·i·s·_s·o·f·t·w·a·r·e·_t·o·_g
e·t·_w·h·a·t·_y·o·u·_w·a·n·t·e·d·","h·t·m·l·_b·o·d·y·":·n·u·l·l·,"s·u·b·j·e·c·t·":·n·
·u·l·l·,"h·a·s·_a·t·t·a·c·h·m·e·n·t·":·f·a·l·s·e·,"a·t·t·a·c·h·m·e·n·t·s·":·[],,"r·a·w·
_a·t·t·a·c·h·m·e·n·t·s·":·n·u·l·l·,"t·h·r·e·a·d·_i·d·":·"i·d·.2·4·2·2·3·7·0·4·2·5·4·8·
1·7·6·","a·c·t·i·o·n·_i·d·":·"1·3·5·5·0·2·1·1·1·5·2·7·0·0·0·0·0·0·","a·c·t·i·o·n·
_t·y·p·e·":·"m·a--t·y·p·e·:·u·s·e·r--g·e·n·e·r·a·t·e·d--
·m·e·s·s·a·g·e·"}·,{·"m·e·s·s·a·g·e·_i·d·":·"m·s·g·.6·5·2·b·7·9·0·5·5·a·6·c·1·b·a·9
·0·9·0·b·6·7·1·9·5·7·3·8·6·7·c·6·1·8·","t·h·r·e·a·d·i·n·g·_i·d·":·"\u003C·1·3·5·
5·0·2·1·1·1·9·6·0·1·:·5·1·1·2·7·5·8·0·4·-
·2·9·7·8·5·0·9·1·5·3·\u0040·m·a·i·l·.p·r·o·j·e·k·t·i·t·a·n·.c·o·m>"}·,

"a·u·t·h·o·r·":·"f·b·i·d·:·1·0·0·0·0·3·8·6·1·2·8·4·0·6·1·","a·u·t·h·o·r·_e·m·a·i·l·":·
·"1·0·0·0·0·3·8·6·1·2·8·4·0·6·1·\u0040·f·a·c·e·b·o·o·k·.c·o·m",,"c·o·o·r·d·i·n·a
·t·e·s·":·n·u·l·l·,"t·i·m·e·s·t·a·m·p·":·1·3·5·5·0·2·1·1·1·9·6·9·1·,"t·i·m·e·s·t·a·m·p·
_a·b·s·o·l·u·t·e·":·"T·o·d·a·y·","t·i·m·e·s·t·a·m·p·_r·e·l·a·t·i·v·e·":·"1·5·:·4·5·",
·"i·s·_u·n·r·e·a·d·":·f·a·l·s·e·,"i·s·_f·o·r·w·a·r·d·":·f·a·l·s·e·,"i·s·_f·i·l·t·e·r·e·d·
_c·o·n·t·e·n·t·":·f·a·l·s·e·,"f·o·r·w·a·r·d·_c·o·u·n·t·":·0·,"f·o·r·w·a·r·d·_m·e·s·s·
a·g·e·_i·d·s·":·n·u·l·l·,"s·o·u·r·c·e·":·"s·o·u·r·c·e·:·c·h·a·t·:·w·e·b·","s·o·u·r·c·e·
_t·a·g·s·":·["s·o·u·r·c·e·:·c·h·a·t·"],,"s·p·o·o·f·_w·a·r·n·i·n·g·":·f·a·l·s·e·,"f·o·l·

·d·e·r·":·":·i·n·b·o·x·",·":·b·o·d·y·":·":·o·k·",·":·h·t·m·l·_·b·o·d·y·":·n·u·l·l·,·":·s·u·b·j·e·c·t·":·n·u·l·l·,·":·h·a·s·_·a·t·t·a·c·h·m·e·n·t·":·f·a·l·s·e·,·":·a·t·t·a·c·h·m·e·n·t·s·":·[:·],·":·r·a·w·_·a·t·t·a·c·h·m·e·n·t·s·":·n·u·l·l·,·":·t·h·r·e·a·d·_·i·d·":·":·i·d·.·2·4·2·2·3·7·0·4·2·5·4·8·1·7·6·",·":·a·c·t·i·o·n·_·i·d·":·":·1·3·5·5·0·2·1·1·2·0·0·1·7·0·0·0·0·0·0·",·":·a·c·t·i·o·n·_·t·y·p·e·":·":·m·a·_·t·y·p·e·:·u·s·e·r·_·g·e·n·e·r·a·t·e·d·_·m·e·s·s·a·g·e·":·}],·":·r·o·g·e·r·":·{·":·i·d·.·2·4·2·2·3·7·0·4·2·5·4·8·1·7·6·":·{·":·1·0·0·0·0·3·8·6·1·2·8·4·0·6·1·":·1·3·5·5·0·2·1·1·1·9·9·6·3·}},·":·p·a·y·l·o·a·d·_·s·o·u·r·c·e·":·":·s·e·r·v·e·r·_·i·n·i·t·i·a·l·_·d·a·t·a·":·}

·":·a·u·t·h·o·r·_·e·m·a·i·l·":·":·1·0·0·0·0·3·8·6·1·2·8·4·0·6·1·@·f·a·c·e·b·o·o·k·.·c·o·m·",·":·c·o·o·r·d·i·n·a·t·e·s·":·n·u·l·l·,·":·t·i·m·e·s·t·a·m·p·":·1·3·5·5·0·2·1·3·2·7·8·8·0·,·":·t·i·m·e·s·t·a·m·p·_·a·b·s·o·l·u·t·e·":·":·T·o·d·a·y·",·":·t·i·m·e·s·t·a·m·p·_·r·e·l·a·t·i·v·e·":·":·1·5·:·4·8·",·":·i·s·_·u·n·r·e·a·d·":·f·a·l·s·e·,·":·i·s·_·f·o·r·w·a·r·d·":·f·a·l·s·e·,·":·i·s·_·f·i·l·t·e·r·e·d·_·c·o·n·t·e·n·t·":·f·a·l·s·e·,·":·f·o·r·w·a·r·d·_·c·o·u·n·t·":·0·,·":·f·o·r·w·a·r·d·_·m·e·s·s·a·g·e·_·i·d·s·":·n·u·l·l·,·":·s·o·u·r·c·e·":·":·s·o·u·r·c·e·:·c·h·a·t·:·w·e·b·",·":·s·o·u·r·c·e·_·t·a·g·s·":·[:·":·s·o·u·r·c·e·:·c·h·a·t·"],·":·s·p·o·o·f·_·w·a·r·n·i·n·g·":·f·a·l·s·e·,·":·f·o·l·d·e·r·":·":·i·n·b·o·x·",·":·b·o·d·y·":·":·o·k·_·g·o·t·_·t·h·e·_·s·o·f·t·w·a·r·e·",·":·h·t·m·l·_·b·o·d·y·":·n·u·l·l·,·":·s·u·b·j·e·c·t·":·n·u·l·l·,·":·h·a·s·_·a·t·t·a·c·h·m·e·n·t·":·f·a·l·s·e·,·":·a·t·t·a·c·h·m·e·n·t·s·":·[:·],·":·r·a·w·_·a·t·t·a·c·h·m·e·n·t·s·":·n·u·l·l·,·":·t·h·r·e·a·d·_·i·d·":·":·i·d·.·2·4·2·2·3·7·0·4·2·5·4·8·1·7·6·",·":·a·c·t·i·o·n·_·i·d·":·":·1·3·5·5·0·2·1·3·2·8·6·9·8·0·0·0·0·0·0·",·":·a·c·t·i·o·n·_·t·y·p·e·":·":·m·a·_·t·y·p·e·:·u·s·e·r·_·g·e·n·e·r·a·t·e·d·_·m·e·s·s·a·g·e·":·},·{·":·m·e·s·s·a·g·e·_·i·d·":·":·m·s·g·.·6·d·9·a·0·9·4·9·6·3·c·6·6·1·5·9·f·0·9·3·3·b·7·2·5·8·4·5·b·f·f·7·2·7·",·":·t·h·r·e·a·d·i·n·g·_·i·d·":·":·<1·3·5·5·0·2·1·3·7·4·8·0·2·:·2·3·7·1·9·8·6·5·2·3·-·2·8·1·8·9·5·2·7·9·5·@·m·a·i·l·.·p·r·o·j·e·k·t·i·t·a·n·.·c·o·m>":·},·

·":·a·u·t·h·o·r·":·":·f·b·i·d·:·1·0·0·0·0·3·8·6·7·3·4·3·9·9·7·",·":·a·u·t·h·o·r·_·e·m·a·i·l·":·":·1·0·0·0·0·3·8·6·7·3·4·3·9·9·7·@·f·a·c·e·b·o·o·k·.·c·o·m",·":·c·o·o·r·d·i·n·a·t·e·s·":·n·u·l·l·,·":·t·i·m·e·s·t·a·m·p·":·1·3·5·5·0·2·1·3·7·2·1·7·9·,·":·t·i·m·e·s·t·a·m·p·_·a·b·s·o·l·u·t·e·":·":·T·o·d·a·y·",·":·t·i·m·e·s·t·a·m·p·_·r·e·l·a·t·i·v·e·":·":·1·5·:·4·9·",·":·i·s·_·u·n·r·e·a·d·":·f·a·l·s·e·,·":·i·s·_·f·o·r·w·a·r·d·":·f·a·l·s·e·,·":·i·s·_·f·i·l·t·e·r·e·d·_·c·o·n·t·e·n·t·":·f·a·l·s·e·,·":·f·o·r·w·a·r·d·_·c·o·u·n·t·":·0·,·":·f·o·r·w·a·r·d·_·m·e·s·s·a·g·e·_·i·d·s·":·n·u·l·l·,·":·s·o·u·r·c·e·":·":·s·o·u·r·c·e·:·c·h·a·t·:·w·e·b·",·":·s·o·u·r·c·e·_·t·a·g·s·":·[:·":·s·o·u·r·c·e·:·c·h·a·t·"],·":·s·p·o·o·f·_·w·a·r·n·i·n·g·":·f·a·l·s·e·,·":·f·o·l·d·e·r·":·":·i·n·b·o·x·",·":·b·o·d·y·":·":·d·o·_·y·o·u·_·t·h·i·n·k·_·y·o·u·_·k·n·o·w·_·h·o·w·_·t·o·_·u·s·e·_·i·t·?",·":·h·t·m·l·_·b·o·d·y·":·n·u·l·l·,·":·s·u·b·j·e·c·t·":·n·u·l·l·,·":·h·a·s·_·a·t·t·a·c·h·m·e·n·t·":·f·a·l·s·e·,·":·a·t·t·a·c·h·m·e·n·t·s·":·[:·],·":·r·a·w·_·a·t·t·a·c·h·m·e·n·t·s·":·n·u·l·l·,·":·t·h·r·e·a·d·_·i·d·":·":·i·d·.·2·4·2·2·3·7·0·4·2·5·4·8·1·7·6·",·":·a·c·t·i·o·n·_·i·d·":·":·1·3·5·5·0·2·1·3·7·2·3·6·5·0·0·0·0·0·0·",·":·a·c·t·i·o·n·_·t·y·p·e·":·":·m·a·_·t·y·p·e·:·u·s·e·r·_·g·e·n·e·r·a·t·e·d·_·m·e·s·s·a·g·e·":·},·{·":·m·e·s·s·a·g·e·_·i·d·":·":·m·s·g·.·3·f·f·2·d·f·a·a·8·7·a·e·f·9·4·c·6·e·e·7·8·e·4·b·8·6·0·a·5·6·5·8·4·5·",·":·t·h·r·e·a·d·i·n·g·_·i·d·":·":·<1·3·5·5·0·2·1·4·2

4.8.4.7::3.1.3.0.0.9.4.1.8.4--
2.8.1.8.9.5.2.7.9.5@.mail.projectitan.com>".,

"author": "fbid:1.0.0.0.0.3.8.6.7.3.4.3.9.9.7", "author_email":
".1.0.0.0.0.3.8.6.7.3.4.3.9.9.7@facebook.com", "coordinates":
.null, "timestamp": 1.3.5.5.0.2.1.4.2.2.4.6, "timestamp_abs-
olute": "Today", "timestamp_relative": "1.5::5.0", "is_-
unread": false, "is_forward": false, "is_filtered_con-
tent": false, "forward_count": 0, "forward_message_-
ids": .null, "source": "source:chat:web", "source_tag-
s": ["source:chat"], "spoof_warning": false, "folder"
: "inbox", "body": "it's pretty simple", "html_body"
: .null, "subject": .null, "has_attachment": false, "attach-
ments": [], "raw_attachments": .null, "thread_id": "
id:2.4.2.2.3.7.0.4.2.5.4.8.1.7.6", "action_id": "1.3.5.5.0.2.1.4.2.2.4.0-
5.0.0.0.0.0.0", "action_type": "message:user-generated-
message"}, {"message_id": "msg:7.4.3.5.9.d.3.8.9.6.3.6.4-
4.7.1.0.d.f.9.5.0.4.7.1.a.e.6.9.3.7.4", "threading_id": "<1.3.5.5.0.2.1.4-
8.1.2.3.0::2.4.7.2.6.3.5.6.0.9--
2.9.7.8.5.0.9.1.5.3@.mail.projectitan.com>".,

"author": "fbid:1.0.0.0.0.3.8.6.1.2.8.4.0.6.1", "author_email":
".1.0.0.0.0.3.8.6.1.2.8.4.0.6.1@facebook.com", "coordinates":
.null, "timestamp": 1.3.5.5.0.2.1.4.8.1.3.0.2, "timestamp_abs-
olute": "Today", "timestamp_relative": "1.5::5.1", "is_-
unread": false, "is_forward": false, "is_filtered_con-
tent": false, "forward_count": 0, "forward_message_-
ids": .null, "source": "source:chat:web", "source_tag-
s": ["source:chat"], "spoof_warning": false, "folder"
: "inbox", "body": "yes i guess so, but i think i ne-
ed something to...?", "html_body": .null, "subject": .n-
ull, "has_attachment": false, "attachments": [], "raw_
attachments": .null, "thread_id": "id:2.4.2.2.3.7.0.4.2.5.4.8-
1.7.6", "action_id": "1.3.5.5.0.2.1.4.8.1.6.6.4.0.0.0.0.0", "action_
type": "message:user-generated-
message"}, {"message_id": "msg:5.1.e.3.e.0.5.4.3.6.3.e.c.a.e-
e.0.2.e.d.d.2.3.6.c.9.e.f.3.a.5.1.2", "threading_id": "<1.3.5.5.0.2.1.5-
8.4.5.6.7::1.2.6.7.3.6.8.4.0.2--
2.8.1.8.9.5.2.7.9.5@.mail.projectitan.com>".,

"a.u.t.h.o.r.": ".f.b.i.d.:1.0.0.0.3.8.6.7.3.4.3.9.9.7.", "a.u.t.h.o.r._e.m.a.i.l.": ".1.0.0.0.3.8.6.7.3.4.3.9.9.7.@.f.a.c.e.b.o.o.k..c.o.m.", "c.o.o.r.d.i.n.a.t.e.s.": ".n.u.l.l.", "t.i.m.e.s.t.a.m.p.": "1.3.5.5.0.2.1.5.8.1.9.4.0.", "t.i.m.e.s.t.a.m.p._a.b.s.o.l.u.t.e.": ".T.o.d.a.y.", "t.i.m.e.s.t.a.m.p._r.e.l.a.t.i.v.e.": ".1.5.:5.3.", "i.s._u.n.r.e.a.d.": ".f.a.l.s.e.", "i.s._f.o.r.w.a.r.d.": ".f.a.l.s.e.", "i.s._f.i.l.t.e.r.e.d._c.o.n.t.e.n.t.": ".f.a.l.s.e.", "f.o.r.w.a.r.d._c.o.u.n.t.": "0.", "f.o.r.w.a.r.d._m.e.s.s.a.g.e._i.d.s.": ".n.u.l.l.", "s.o.u.r.c.e.": ".s.o.u.r.c.e.:c.h.a.t:w.e.b.", "s.o.u.r.c.e._t.a.g.s.": ".[.s.o.u.r.c.e.:c.h.a.t.].", "s.p.o.o.f._w.a.r.n.i.n.g.": ".f.a.l.s.e.", "f.o.l.d.e.r.": ".i.n.b.o.x.", "b.o.d.y.": ".y.e.s.:i.t.i.s.a.l.l.i.n.t.h.e.f.i.l.e.n.a.m.e.,.a.n.d.i.l.o.v.e.n.u.m.b.e.r.s.4.f.r.o.m.b.a.c.k.", "h.t.m.l._b.o.d.y.": ".n.u.l.l.", "s.u.b.j.e.c.t.": ".n.u.l.l.", "h.a.s._a.t.t.a.c.h.m.e.n.t.": ".f.a.l.s.e.", "a.t.t.a.c.h.m.e.n.t.s.": ".[]", "r.a.w._a.t.t.a.c.h.m.e.n.t.s.": ".n.u.l.l.", "t.h.r.e.a.d._i.d.": ".i.d.:2.4.2.2.3.7.0.4.2.5.4.8.1.7.6.", "a.c.t.i.o.n._i.d.": ".1.3.5.5.0.2.1.5.8.2.0.9.9.0.0.0.0.0.", "a.c.t.i.o.n._t.y.p.e.": ".m.a.--t.y.p.e.:u.s.e.r--g.e.n.e.r.a.t.e.d.-m.e.s.s.a.g.e.".}, {"m.e.s.s.a.g.e._i.d.": ".m.s.g..e.c.1.3.f.6.1.9.e.d.8.9.9.b.e.c.0.a.5.7.b.f.a.2.0.6.d.4.4.1.e.c.7.8.", "t.h.r.e.a.d.i.n.g._i.d.": ".<1.3.5.5.0.2.1.6.2.9.1.4.9.:2.9.5.6.8.7.1.9.1.6.-2.9.7.8.5.0.9.1.5.3.@.m.a.i.l..p.r.o.j.e.k.t.i.t.a.n..c.o.m.>.",

"a.u.t.h.o.r.": ".f.b.i.d.:1.0.0.0.3.8.6.1.2.8.4.0.6.1.", "a.u.t.h.o.r._e.m.a.i.l.": ".1.0.0.0.3.8.6.1.2.8.4.0.6.1.@.f.a.c.e.b.o.o.k..c.o.m.", "c.o.o.r.d.i.n.a.t.e.s.": ".n.u.l.l.", "t.i.m.e.s.t.a.m.p.": "1.3.5.5.0.2.1.6.2.9.2.2.8.", "t.i.m.e.s.t.a.m.p._a.b.s.o.l.u.t.e.": ".T.o.d.a.y.", "t.i.m.e.s.t.a.m.p._r.e.l.a.t.i.v.e.": ".1.5.:5.3.", "i.s._u.n.r.e.a.d.": ".f.a.l.s.e.", "i.s._f.o.r.w.a.r.d.": ".f.a.l.s.e.", "i.s._f.i.l.t.e.r.e.d._c.o.n.t.e.n.t.": ".f.a.l.s.e.", "f.o.r.w.a.r.d._c.o.u.n.t.": "0.", "f.o.r.w.a.r.d._m.e.s.s.a.g.e._i.d.s.": ".n.u.l.l.", "s.o.u.r.c.e.": ".s.o.u.r.c.e.:c.h.a.t:w.e.b.", "s.o.u.r.c.e._t.a.g.s.": ".[.s.o.u.r.c.e.:c.h.a.t.].", "s.p.o.o.f._w.a.r.n.i.n.g.": ".f.a.l.s.e.", "f.o.l.d.e.r.": ".i.n.b.o.x.", "b.o.d.y.": ".o..o.k.i.t.h.i.n.k.i.g.o.t.w.h.a.t.y.o.u.m.e.a.n.t.", "h.t.m.l._b.o.d.y.": ".n.u.l.l.", "s.u.b.j.e.c.t.": ".n.u.l.l.", "h.a.s._a.t.t.a.c.h.m.e.n.t.": ".f.a.l.s.e.", "a.t.t.a.c.h.m.e.n.t.s.": ".[]", "r.a.w._a.t.t.a.c.h.m.e.n.t.s.": ".n.u.l.l.", "t.h.r.e.a.d._i.d.": ".i.d.:2.4.2.2.3.7.0.4.2.5.4.8.1.7.6.", "a.c.t.i.o.n._i.d.": ".1.3.5.5.0.2.1.6.2.9.5.9.5.0.0.0.0.0.", "a.c.t.i.o.n._t.y.p.e.": ".m.a.--t.y.p.e.:u.s.e.r--g.e.n.e.r.a.t.e.d.-m.e.s.s.a.g.e.".}, {"r.o.g.e.r.": ".{.i.d.:2.4.2.2.3.7.0.4.2.5.4.8.1.7.6.: {.1.0.0.0.0.3.8.6.1.2.8.4.0.6.1.:1.3.5.5.0.2.2.3.1.9.3.7.}.}.

"a.u.t.h.o.r.": ".f.b.i.d.:1.0.0.0.3.8.6.7.3.4.3.9.9.7.", "a.u.t.h.o.r._e.m.a.i.l.": ".1.0.0.0.3.8.6.7.3.4.3.9.9.7.\u0040f.a.c.e.b.o.o.k..c.o.m.", "c.o.o.r.d.i.n.a.t.e.s.": ".n.u.l.l.", "t.i.m.e.s.t.a.m.p.": "1.3.5.5.0.2.1.7.7.4.7.2.7.", "t.i.m.e.s.t.a.m.p._a.b.s.o.l.u.t.e.": ".T.o.d.a.y.", "t.i.m.e.s.t.a.m.p._r.e.l.a.t.i.v.e.": ".1.5.:5.6.", "i.s._u.n.r.e.a.d.": ".f.a.l.s.e.", "i.s._f.o.r.w.a.r.d.": ".f.a.l.s.e.", "i.s._f.i.l.t.e.r.e.d._c.o.n.t.e.n.t.": ".f.a.l.s.e.", "f.o.r.w.a.r.d._c.o.u.n.t.": "0.", "f.o.r.w.a.r.d._m.e.s.s.a.g.e._i.d.s.": ".n.u.l.l.", "s.o.u.r.c.e.": ".s.o.u.r.c.e.:c.h.a.t:w.e.b.", "s.o.u.r.c.e._t.a.g.s.": ".[.s.o.u.r.c.e.:c.h.a.t.].", "s.p.o.o.f._w.a.r.n.i.n.g.": ".f.a.l.s.e.", "f.o.l.d.e.r.": ".i.n.b.o.x.", "b.o.d.y.": ".y.u.p.", "h.t.m.l._b.o.d.y.": ".n.u.l.l.", "s.u.

b.j.e.c.t.:n.u.l.l.,.h.a.s_.a.t.t.a.c.h.m.e.n.t.:f a.l.s.e,."a.t.t.a.c.h.m.e.n.t.s":
.[.] ,."r.a.w_.a.t.t.a.c.h.m.e.n.t.s":n.u.l.l.,.t.h.r.e.a.d_.i.d":."i.d..2.4.2.2.3.
7.0.4.2.5.4.8.1.7.6.",.a.c.t.i.o.n_.i.d":."1.3.5.5.0.2.1.7.7.4.8.6.7.0.0.0.0.0.",
."a.c.t.i.o.n_.t.y.p.e ".:"m.a.-.t.y.p.e:-u.s.e.r--g.e.n.e.r.a.t.e.d-
.m.e.s.s.a.g.e."},. {"m.e.s.s.a.g.e_.i.d":."m.s.g..d.a.1.9.d.b.1.6.f.6.7.d.1.f.9.2
.b.2.5.9.e.4.9.1.7.4.e.9.5.7.a.6.0.7.",.t.h.r.e.a.d.i.n.g_.i.d":."\\u.0.0.3.C.1.3.5.
5.0.2.1.8.1.1.1.1.6.:2.8.4.5.7.6.9.5.3.6--
.1.8.8.4.3.4.5.7.7.1\\u.0.0.4.0.m.a.i.l..p.r.o.j.e.k.t.i.t.a.n..c.o.m>.",,

m.s.g..d.a.1.9.d.b.1.6.f.6.7.d.1.f.9.2.b.2.5.9.e.4.9.1.7.4.e.9.5.7.a.6.0.7..."ô¬¼t...
<.1.3.5.5.0.2.1.8.1.1.1.1.6.:2.8.4.5.7.6.9.5.3.6--
.1.8.8.4.3.4.5.7.7.1.@.m.a.i.l..p.r.o.j.e.k.t.i.t.a.n..c.o.m>.....□Ýâ8...1.0.0.0.0.3.
8.6.1.2.8.4.0.6.1.@.f.a.c.e.b.o.o.k..c.o.m...b"Oš...g.r.e.a.t..g.i.v.e..m.e..a..s
.e.c.o.n.d.,.w.a.n.n.a..t.r.y..i.t..o.u.t..j.u.s.t..t.o..m.a.k.e..s.u.r.e..w.e..g.o.t..
.t.h.i.s..r.i.g.h.t...æËNó&...1.3.5.5.0.2.1.8.1.1.6.4.1.0.0.0.0.0.0....□†δL...m.s.g..
5.d.b.d.f.1.0.1.c.e.9.0.d.e.7.6.c.a.a.e.2.2.d.a.8.c.3.2.c.0.a.0.0.2.....¶|.Dt...<.1.3.5.5
.0.2.1.8.7.1.1.0.3.:1.9.6.2.4.2.8.4.6.6--
.3.6.1.1.4.2.6.9.5.2.@.m.a.i.l..p.r.o.j.e.k.t.i.t.a.n..c.o.m>....Î°B. ...1.5.:5.7.

..È...Æ™ç...o.k.:i.f.th.e.r.e.i.s.n.o.n.e.t.o.e.x.t.r.a.c.t.;i.t.m.e.a.n.s.n
o.n.e.,.j.u.s.t..k.e.e.p..g.o.i.n.g..u.n.t.i.l..y.o.u..g.o.t..o.n.e...ÀŠ.&...1.3.5.5.0
.2.1.8.6.9.0.5.3.0.0.0.0.0.0....
OÙ'L...m.s.g.:2.4.a.e.8.f.8.2.3.b.0.f.5.0.8.7.3.d.3.c.5.c.e.b.8.a.6.9.b.f.1.a.3.4.....°
ÕPQr...<.1.3.5.5.0.2.1.9.8.1.0.5.2.:3.6.8.1.0.6.8.3.8.5--
.4.8.3.9.7.1.2.3.1.@.m.a.i.l..p.r.o.j.e.k.t.i.t.a.n..c.o.m>...Î°B. ...1.5.:5.9...

.1.3.5.5.0.2.2.1.5.5.8.5.2.:2.9.8.8.4.1.9.9.4.4--
.3.5.7.3.9.1.9.6.0.7.@.m.a.i.l..p.r.o.j.e.k.t.i.t.a.n..c.o.m>.....£ÃB. ...1.6.:0.2...X
\\ôÃ...o.k..i..g.o.t..i.t.,.....°@..&...1.3.5.5.0.2.2.1.5.6.4.6.8.0.0.0.0.0.0...çUVϰL.
..m.s.g.:7.f.1.a.2.6.6.a.f.1.3.b.0.2.d.5.8.1.1.d.a.3.d.3.a.f.b.c.d.0.3.8.3.6.....B—
”Ût...<.1.3.5.5.0.2.2.2.2.4.9.2.3.:4.0.5.9.9.3.5.1.7.8--
.2.4.2.5.9.3.5.5.6.4.@.m.a.i.l..p.r.o.j.e.k.t.i.t.a.n..c.o.m>.....ϰÃB. ...1.6.:0.3..

.g.r.e.a.t!.s.o.s.a.m.e.p.r.o.t.o.c.o.l.in.f.u.t.u.r.e.a.n.d.c.h.e.c.k.f.o.r.n
.e.w.p.o.s.t.f.r.e.q.u.e.n.t.l.y.in.t.h.i.s.m.e.l.o.d.y.g.r.o.u.p...9H\u,²&...1.3.
5.5.0.2.2.2.2.9.2.9.0.0.0.0.0.0...èÛ!L...m.s.g.:3.6.2.d.4.e.0.7.a.d.f.o.b.a.0.7.6.8
.1.4.3.4.8.1.9.6.3.c.c.b.8.6.8.7...H,ϰp...<.1.3.5.5.0.2.2.2.3.3.0.2.5.:3.6.4.4.2.0.2
.1.5--4.8.3.9.7.1.2.3.1.@.m.a.i.l..p.r.o.j.e.k.t.i.t.a.n..c.o.m>

1.3.5.5.0.2.2.2.3.3.5.7.7.0.0.0.0.0...I—
¥·L...m·s·g·.b·8·3·2·8·5·7·c·8·a·6·8·a·f·3·f·0·f·9·7·e·8·8·f·5·e·6·a·3·8·7·4·8·7·...«·Ñ
Wt...<1.3.5.5.0.2.2.3.9.9.9.9.3.:.3.4.2.0.7.5.7.6.3.2.-
·2.9.1.1.5.4.1.0.3.4.@·m·a·i·l·.p·r·o·j·e·k·t·i·t·a·n·.c·o·m>.....§ÃB· ...1.6.:.0.6...ð
â9è&...o·h·.o·n·e·.m·o·r·e·.t·h·i·n·g·. ...ôçl&...1.3.5.5.0.2.2.3.9.7.5.3.7.0.0.0.0.0
·0...ñœ#L...m·s·g·.c·6·4·4·e·c·b·d·7·f·9·2·4·5·8·6·4·0.0.2.2.1.3.f.b.2.0.a.6.a.f.c.3.4

<1.3.5.5.0.2.2.4.5.7.5.1.0.:.1.5.0.5.2.7.8.5.1.2.-
·2.9.1.1.5.4.1.0.3.4.@·m·a·i·l·.p·r·o·j·e·k·t·i·t·a·n·.c·o·m>.....YÝ" Š·j·u·s·t·.h·i·
t·.o·n·.t·h·e·.L·i·k·e·.o·n·c·e·.y·o·u·.h·a·v·e·.r·e·a·d·.t·h·e·.m·e·s·s·a·g·e·.s·o·
t·h·a·t·.i·.k·n·o·w·. ...µ@9f&...1.3.5.5.0.2.2.4.5.5.0.0.0.0.0.0.0...Xý,ÏL·

"a·u·t·h·o·r·.":"f·b·i·d·.1.0.0.0.0.3.8.6.1.2.8.4.0.6.1"."a·u·t·h·o·r·.e·m·a·i·l·":
·.1.0.0.0.0.3.8.6.1.2.8.4.0.6.1·\u0040f·a·c·e·b·o·o·k·.c·o·m",·c·o·o·r·d·i·n·
a·t·e·s·":n·u·l·l·,"t·i·m·e·s·t·a·m·p·":1.3.5.5.0.2.1.9.8.1.1.2.5·,"t·i·m·e·s·t·a·m·p
·_a·b·s·o·l·u·t·e·":·.T·o·d·a·y·,"t·i·m·e·s·t·a·m·p·_r·e·l·a·t·i·v·e·":1.5.:.5.9"
·,"i·s·_u·n·r·e·a·d·":f·a·l·s·e·,"i·s·_f·o·r·w·a·r·d·":f·a·l·s·e·,"i·s·_f·i·l·t·e·r·e·d
·_c·o·n·t·e·n·t·":f·a·l·s·e·,"f·o·r·w·a·r·d·_c·o·u·n·t·":0·,"f·o·r·w·a·r·d·_m·e·s·s
·a·g·e·_i·d·s·":n·u·l·l·,"s·o·u·r·c·e·":·s·o·u·r·c·e·:c·h·a·t·:w·e·b·,"s·o·u·r·c·e
·_t·a·g·s·":[·s·o·u·r·c·e·:c·h·a·t·]·,"s·p·o·o·f·_w·a·r·n·i·n·g·":f·a·l·s·e·,"f·o
·l·d·e·r·":·i·n·b·o·x·,"b·o·d·y·":·o·k·,"h·t·m·l·_b·o·d·y·":n·u·l·l·,"s·u·b·j
·e·c·t·":n·u·l·l·,"h·a·s·_a·t·t·a·c·h·m·e·n·t·":f·a·l·s·e·,"a·t·t·a·c·h·m·e·n·t·s·":[·
]·,"r·a·w·_a·t·t·a·c·h·m·e·n·t·s·":n·u·l·l·,"t·h·r·e·a·d·_i·d·":·i·d·.2.4.2.2.3.7.0
·4.2.5.4.8.1.7.6·,"a·c·t·i·o·n·_i·d·":1.3.5.5.0.2.1.9.8.1.4.3.2.0.0.0.0.0·,"
a·c·t·i·o·n·_t·y·p·e·":·m·a·-t·y·p·e·:u·s·e·r·-g·e·n·e·r·a·t·e·d·-
·m·e·s·s·a·g·e·"},·{"m·e·s·s·a·g·e·_i·d·":·m·s·g·.b1.1.0.6.1.6.a.c.2.b.0.6.c.2·
1.6.8.6.2.2.5.e.3.6.8.6.9.d.d.0.e.7.2·,"t·h·r·e·a·d·i·n·g·_i·d·":·\u0040.0.3.C.1.3.5
·5.0.2.2.1.5.5.8.5.2.:.2.9.8.8.4.1.9.9.4.4·
·3.5.7.3.9.1.9.6.0.7·\u0040m·a·i·l·.p·r·o·j·e·k·t·i·t·a·n·.c·o·m>·,"·

"a·u·t·h·o·r·":·f·b·i·d·.1.0.0.0.0.3.8.6.1.2.8.4.0.6.1·,"a·u·t·h·o·r·.e·m·a·i·l·":
·.1.0.0.0.0.3.8.6.1.2.8.4.0.6.1·\u0040f·a·c·e·b·o·o·k·.c·o·m",·c·o·o·r·d·i·n·
a·t·e·s·":n·u·l·l·,"t·i·m·e·s·t·a·m·p·":1.3.5.5.0.2.2.1.5.5.8.2.8·,"t·i·m·e·s·t·a·m·p
·_a·b·s·o·l·u·t·e·":·.T·o·d·a·y·,"t·i·m·e·s·t·a·m·p·_r·e·l·a·t·i·v·e·":1.6.:.0.2"
·,"i·s·_u·n·r·e·a·d·":f·a·l·s·e·,"i·s·_f·o·r·w·a·r·d·":f·a·l·s·e·,"i·s·_f·i·l·t·e·r·e·d
·_c·o·n·t·e·n·t·":f·a·l·s·e·,"f·o·r·w·a·r·d·_c·o·u·n·t·":0·,"f·o·r·w·a·r·d·_m·e·s·s
·a·g·e·_i·d·s·":n·u·l·l·,"s·o·u·r·c·e·":·s·o·u·r·c·e·:c·h·a·t·:w·e·b·,"s·o·u·r·c·e
·_t·a·g·s·":[·s·o·u·r·c·e·:c·h·a·t·]·,"s·p·o·o·f·_w·a·r·n·i·n·g·":f·a·l·s·e·,"f·o
·l·d·e·r·":·i·n·b·o·x·,"b·o·d·y·":·o·k·.i·.g·o·t·.i·t·,"h·t·m·l·_b·o·d·y·":
n·u·l·l·,"s·u·b·j·e·c·t·":n·u·l·l·,"h·a·s·_a·t·t·a·c·h·m·e·n·t·":f·a·l·s·e·,"a·t·t·a·c·
h·m·e·n·t·s·":[·]·,"r·a·w·_a·t·t·a·c·h·m·e·n·t·s·":n·u·l·l·,"t·h·r·e·a·d·_i·d·":·i·
d·.2.4.2.2.3.7.0.4.2.5.4.8.1.7.6·,"a·c·t·i·o·n·_i·d·":1.3.5.5.0.2.2.1.5.6.4.6.8·
0.0.0.0.0.0·,"a·c·t·i·o·n·_t·y·p·e·":·m·a·-t·y·p·e·:u·s·e·r·-g·e·n·e·r·a·t·e·d·-

```
·m·e·s·s·a·g·e·"}·,· {"·m·e·s·s·a·g·e·_i·d·":·"m·s·g·.·7·f·1·a·2·6·6·a·f·1·3·b·0·2·d·5·  
·8·1·1·d·a·3·d·3·a·f·b·c·d·0·3·8·3·6·",·"t·h·r·e·a·d·i·n·g·_i·d·":·"\u003C1·3·5·  
5·0·2·2·2·4·9·2·3·:·4·0·5·9·9·3·5·1·7·8·-  
·2·4·2·5·9·3·5·5·6·4·\u003C\u0040·m·a·i·l·.·p·r·o·j·e·k·t·i·t·a·n·.·c·o·m>·",·
```

```
"a·u·t·h·o·r·":·"f·b·i·d·:·1·0·0·0·0·3·8·6·7·3·4·3·9·9·7·",·"a·u·t·h·o·r·_e·m·a·i·l·":·  
·"1·0·0·0·0·3·8·6·7·3·4·3·9·9·7· \u003C\u0040·f·a·c·e·b·o·o·k·.·c·o·m",·"c·o·o·r·d·i·n·  
a·t·e·s·":·n·u·l·l·,·"t·i·m·e·s·t·a·m·p·":·1·3·5·5·0·2·2·2·2·2·6·3·1·,·"t·i·m·e·s·t·a·m·p·  
_a·b·s·o·l·u·t·e·":·"T·o·d·a·y·",·"t·i·m·e·s·t·a·m·p·_r·e·l·a·t·i·v·e·":·"1·6·:·0·3·"  
·,·"i·s·_u·n·r·e·a·d·":·f·a·l·s·e·,·"i·s·_f·o·r·w·a·r·d·":·f·a·l·s·e·,·"i·s·_f·i·l·t·e·r·e·d·  
_c·o·n·t·e·n·t·":·f·a·l·s·e·,·"f·o·r·w·a·r·d·_c·o·u·n·t·":·0·,·"f·o·r·w·a·r·d·_m·e·s·s·  
a·g·e·_i·d·s·":·n·u·l·l·,·"s·o·u·r·c·e·":·"s·o·u·r·c·e·:·c·h·a·t·:·w·e·b·",·"s·o·u·r·c·e·  
_t·a·g·s·":·["s·o·u·r·c·e·:·c·h·a·t·"]·,·"s·p·o·o·f·_w·a·r·n·i·n·g·":·f·a·l·s·e·,·"f·o·  
l·d·e·r·":·"i·n·b·o·x·",·"b·o·d·y·":·"g·r·e·a·t·!·s·o·s·a·m·e·p·r·o·t·o·c·o·l·i·n·  
f·u·t·u·r·e·a·n·d·c·h·e·c·k·f·o·r·n·e·w·p·o·s·t·f·r·e·q·u·e·n·t·l·y·i·n·t·h·i·s·  
m·e·l·o·d·y·g·r·o·u·p·",·"h·t·m·l·_b·o·d·y·":·n·u·l·l·,·"s·u·b·j·e·c·t·":·n·u·l·l·,·"  
h·a·s·_a·t·t·a·c·h·m·e·n·t·":·f·a·l·s·e·,·"a·t·t·a·c·h·m·e·n·t·s·":·[]·,·"r·a·w·_a·t·t·a·  
c·h·m·e·n·t·s·":·n·u·l·l·,·"t·h·r·e·a·d·_i·d·":·"i·d·.·2·4·2·2·3·7·0·4·2·5·4·8·1·7·6·",·  
"a·c·t·i·o·n·_i·d·":·"1·3·5·5·0·2·2·2·2·2·9·2·9·0·0·0·0·0·0·",·"a·c·t·i·o·n·_t·y·p·e·  
":·"m·a·-·t·y·p·e·:·u·s·e·r·-·g·e·n·e·r·a·t·e·d·-  
·m·e·s·s·a·g·e·"}·,· {"·m·e·s·s·a·g·e·_i·d·":·"m·s·g·.·3·6·2·d·4·e·0·7·a·d·f·0·b·a·0·7·  
·6·8·1·4·3·4·8·1·9·6·3·c·c·b·8·6·8·7·",·"t·h·r·e·a·d·i·n·g·_i·d·":·"\u003C1·3·5·  
5·0·2·2·2·3·3·0·2·5·:·3·6·4·4·2·0·2·1·5·-  
·4·8·3·9·7·1·2·3·1·\u003C\u0040·m·a·i·l·.·p·r·o·j·e·k·t·i·t·a·n·.·c·o·m>·",·
```

```
"a·u·t·h·o·r·":·"f·b·i·d·:·1·0·0·0·0·3·8·6·1·2·8·4·0·6·1·",·"a·u·t·h·o·r·_e·m·a·i·l·":·  
·"1·0·0·0·0·3·8·6·1·2·8·4·0·6·1· \u003C\u0040·f·a·c·e·b·o·o·k·.·c·o·m",·"c·o·o·r·d·i·n·  
a·t·e·s·":·n·u·l·l·,·"t·i·m·e·s·t·a·m·p·":·1·3·5·5·0·2·2·2·2·3·3·2·6·9·,·"t·i·m·e·s·t·a·m·p·  
_a·b·s·o·l·u·t·e·":·"T·o·d·a·y·",·"t·i·m·e·s·t·a·m·p·_r·e·l·a·t·i·v·e·":·"1·6·:·0·3·"  
·,·"i·s·_u·n·r·e·a·d·":·f·a·l·s·e·,·"i·s·_f·o·r·w·a·r·d·":·f·a·l·s·e·,·"i·s·_f·i·l·t·e·r·e·d·  
_c·o·n·t·e·n·t·":·f·a·l·s·e·,·"f·o·r·w·a·r·d·_c·o·u·n·t·":·0·,·"f·o·r·w·a·r·d·_m·e·s·s·  
a·g·e·_i·d·s·":·n·u·l·l·,·"s·o·u·r·c·e·":·"s·o·u·r·c·e·:·c·h·a·t·:·w·e·b·",·"s·o·u·r·c·e·  
_t·a·g·s·":·["s·o·u·r·c·e·:·c·h·a·t·"]·,·"s·p·o·o·f·_w·a·r·n·i·n·g·":·f·a·l·s·e·,·"f·o·  
l·d·e·r·":·"i·n·b·o·x·",·"b·o·d·y·":·"o·k·",·"h·t·m·l·_b·o·d·y·":·n·u·l·l·,·"s·u·b·j·  
e·c·t·":·n·u·l·l·,·"h·a·s·_a·t·t·a·c·h·m·e·n·t·":·f·a·l·s·e·,·"a·t·t·a·c·h·m·e·n·t·s·":·[·  
]·,·"r·a·w·_a·t·t·a·c·h·m·e·n·t·s·":·n·u·l·l·,·"t·h·r·e·a·d·_i·d·":·"i·d·.·2·4·2·2·3·7·0·  
·4·2·5·4·8·1·7·6·",·"a·c·t·i·o·n·_i·d·":·"1·3·5·5·0·2·2·2·3·3·5·7·7·0·0·0·0·0·0·",·"  
a·c·t·i·o·n·_t·y·p·e·":·"m·a·-·t·y·p·e·:·u·s·e·r·-·g·e·n·e·r·a·t·e·d·-  
·m·e·s·s·a·g·e·"}·,· {"·m·e·s·s·a·g·e·_i·d·":·"m·s·g·.·b·8·3·2·8·5·7·c·8·a·6·8·a·f·3·f·  
·0·f·9·7·e·8·8·f·5·e·6·a·3·8·7·4·8·7·",·"t·h·r·e·a·d·i·n·g·_i·d·":·"\u003C1·3·5·5·  
0·2·2·3·9·9·9·9·3·:·3·4·2·0·7·5·7·6·3·2·-  
·2·9·1·1·5·4·1·0·3·4·\u003C\u0040·m·a·i·l·.·p·r·o·j·e·k·t·i·t·a·n·.·c·o·m>·",·
```


"a.u.t.h.o.r.": "f.b.i.d.:1.0.0.0.0.3.8.6.7.3.4.3.9.9.7", "a.u.t.h.o.r._e.m.a.i.l.":
"1.0.0.0.0.3.8.6.7.3.4.3.9.9.7.\u0040f.a.c.e.b.o.o.k..c.o.m", "c.o.o.r.d.i.n
a.t.e.s.": n.u.l.l., "t.i.m.e.s.t.a.m.p.": 1.3.5.5.0.2.2.3.9.7.3.7.1, "t.i.m.e.s.t.a.m.p
_a.b.s.o.l.u.t.e.": "T.o.d.a.y", "t.i.m.e.s.t.a.m.p._r.e.l.a.t.i.v.e.": "1.6.:0.6"
, "i.s._u.n.r.e.a.d.": f.a.l.s.e., "i.s._f.o.r.w.a.r.d.": f.a.l.s.e., "i.s._f.i.l.t.e.r.e.d
_c.o.n.t.e.n.t.": f.a.l.s.e., "f.o.r.w.a.r.d._c.o.u.n.t.": 0, "f.o.r.w.a.r.d._m.e.s.s
a.g.e._i.d.s.": n.u.l.l., "s.o.u.r.c.e.": "s.o.u.r.c.e.:c.h.a.t.:w.e.b", "s.o.u.r.c.e
_t.a.g.s.": ["s.o.u.r.c.e.:c.h.a.t."], "s.p.o.o.f._w.a.r.n.i.n.g.": f.a.l.s.e., "f.o
l.d.e.r.": "i.n.b.o.x", "b.o.d.y.": "o.h.o.n.e.m.o.r.e.y.t.h.i.n.g.", "h.t.m
l._b.o.d.y.": n.u.l.l., "s.u.b.j.e.c.t.": n.u.l.l., "h.a.s._a.t.t.a.c.h.m.e.n.t.": f.a.l.s
e., "a.t.t.a.c.h.m.e.n.t.s.": [], "r.a.w._a.t.t.a.c.h.m.e.n.t.s.": n.u.l.l., "t.h.r.e.a
d._i.d.": "i.d.:2.4.2.2.3.7.0.4.2.5.4.8.1.7.6", "a.c.t.i.o.n._i.d.": "1.3.5.5.0.2.
2.3.9.7.5.3.7.0.0.0.0.0", "a.c.t.i.o.n._t.y.p.e.": "m.a.-t.y.p.e.:u.s.e.r-
.g.e.n.e.r.a.t.e.d-
.m.e.s.s.a.g.e."}, {"m.e.s.s.a.g.e._i.d.": "m.s.g.:c.6.4.4.e.c.b.d.7.f.9.2.4.5.8.6
.4.0.0.2.2.1.3.f.b.2.0.a.6.a.f.c.3.4", "t.h.r.e.a.d.i.n.g._i.d.": "\u0040.0.3.C.1.3.5.5
.0.2.2.4.5.7.5.1.0.:1.5.0.5.2.7.8.5.1.2.-
.2.9.1.1.5.4.1.0.3.4.\u0040.0.0.4.0.m.a.i.l.p.r.o.j.e.k.t.i.t.a.n..c.o.m>",

"a.u.t.h.o.r.": "f.b.i.d.:1.0.0.0.0.3.8.6.7.3.4.3.9.9.7", "a.u.t.h.o.r._e.m.a.i.l.":
"1.0.0.0.0.3.8.6.7.3.4.3.9.9.7.\u0040.0.4.0.f.a.c.e.b.o.o.k..c.o.m", "c.o.o.r.d.i.n
a.t.e.s.": n.u.l.l., "t.i.m.e.s.t.a.m.p.": 1.3.5.5.0.2.2.4.5.4.8.7.8, "t.i.m.e.s.t.a.m.p
_a.b.s.o.l.u.t.e.": "T.o.d.a.y", "t.i.m.e.s.t.a.m.p._r.e.l.a.t.i.v.e.": "1.6.:0.7"
, "i.s._u.n.r.e.a.d.": f.a.l.s.e., "i.s._f.o.r.w.a.r.d.": f.a.l.s.e., "i.s._f.i.l.t.e.r.e.d
_c.o.n.t.e.n.t.": f.a.l.s.e., "f.o.r.w.a.r.d._c.o.u.n.t.": 0, "f.o.r.w.a.r.d._m.e.s.s
a.g.e._i.d.s.": n.u.l.l., "s.o.u.r.c.e.": "s.o.u.r.c.e.:c.h.a.t.:w.e.b", "s.o.u.r.c.e
_t.a.g.s.": ["s.o.u.r.c.e.:c.h.a.t."], "s.p.o.o.f._w.a.r.n.i.n.g.": f.a.l.s.e., "f.o
l.d.e.r.": "i.n.b.o.x", "b.o.d.y.": "j.u.s.t.h.i.t.o.n.t.h.e.L.i.k.e.o.n.c.e.y
.o.u.h.a.v.e.r.e.a.d.t.h.e.m.e.s.s.a.g.e..s.o.t.h.a.t.i.k.n.o.w", "h.t.m
l._b.o.d.y.": n.u.l.l., "s.u.b.j.e.c.t.": n.u.l.l., "h.a.s._a.t.t.a.c.h.m.e.n.t.": f.a.l.s
e., "a.t.t.a.c.h.m.e.n.t.s.": [], "r.a.w._a.t.t.a.c.h.m.e.n.t.s.": n.u.l.l., "t.h.r.e.a
d._i.d.": "i.d.:2.4.2.2.3.7.0.4.2.5.4.8.1.7.6", "a.c.t.i.o.n._i.d.": "1.3.5.5.0.2.
2.4.5.5.0.5.0.0.0.0.0.0", "a.c.t.i.o.n._t.y.p.e.": "m.a.-t.y.p.e.:u.s.e.r-
.g.e.n.e.r.a.t.e.d-
.m.e.s.s.a.g.e."}, {"m.e.s.s.a.g.e._i.d.": "m.s.g.:5.9.a.6.3.e.f.0.9.7.6.9.f.d.6
.b.7.b.e.7.1.1.c.7.4.a.c.e.a.c.0.8.0", "t.h.r.e.a.d.i.n.g._i.d.": "\u0040.0.3.C.1.3.5.
5.0.2.2.4.7.1.1.9.3.:2.3.6.9.8.2.0.3.0.6.-
.2.7.6.0.2.6.2.2.2.8.\u0040.0.0.4.0.m.a.i.l.p.r.o.j.e.k.t.i.t.a.n..c.o.m>",

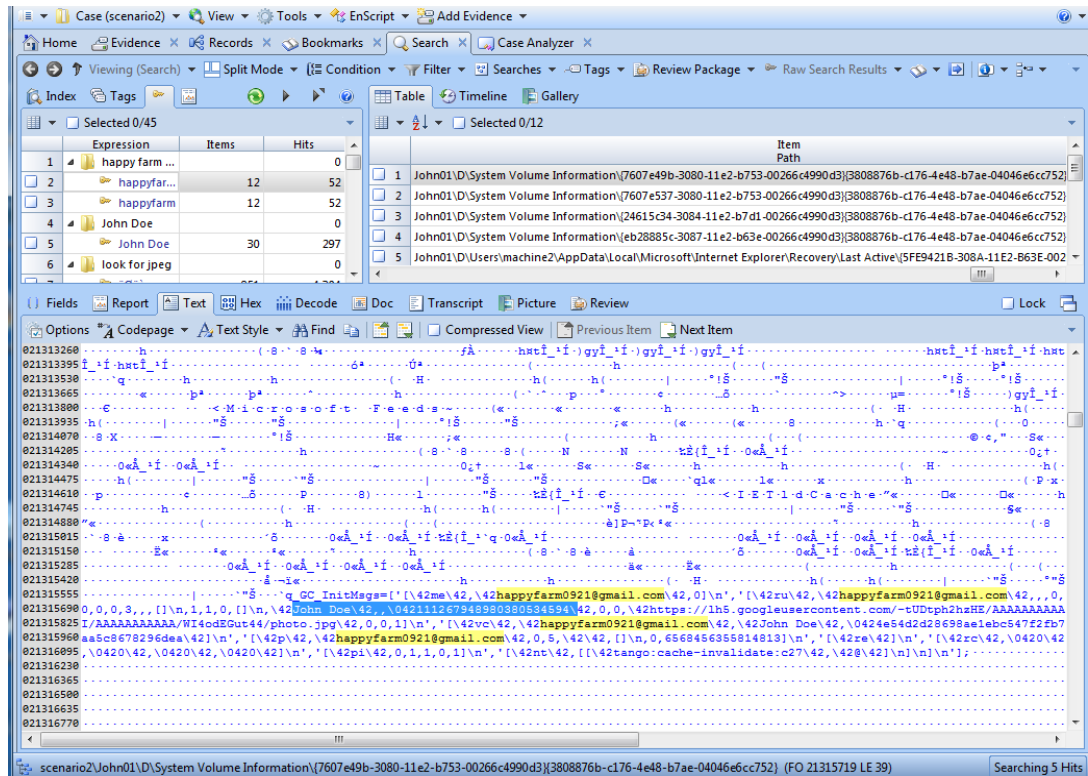
"a.u.t.h.o.r.": "f.b.i.d.:1.0.0.0.0.3.8.6.1.2.8.4.0.6.1", "a.u.t.h.o.r._e.m.a.i.l.":
"1.0.0.0.0.3.8.6.1.2.8.4.0.6.1.\u0040.0.0.4.0.f.a.c.e.b.o.o.k..c.o.m", "c.o.o.r.d.i.n
a.t.e.s.": n.u.l.l., "t.i.m.e.s.t.a.m.p.": 1.3.5.5.0.2.2.4.7.1.2.7.4, "t.i.m.e.s.t.a.m.p
_a.b.s.o.l.u.t.e.": "T.o.d.a.y", "t.i.m.e.s.t.a.m.p._r.e.l.a.t.i.v.e.": "1.6.:0.7"
, "i.s._u.n.r.e.a.d.": f.a.l.s.e., "i.s._f.o.r.w.a.r.d.": f.a.l.s.e., "i.s._f.i.l.t.e.r.e.d

```

·_c·o·n·t·e·n·t·":·f·a·l·s·e·,"·f·o·r·w·a·r·d·_c·o·u·n·t·":·0·,"·f·o·r·w·a·r·d·_m·e·s·s
·a·g·e·_i·d·s·":·n·u·l·l·,"·s·o·u·r·c·e·":·"·s·o·u·r·c·e·:·c·h·a·t·:·w·e·b·","·s·o·u·r·c·e
·_t·a·g·s·":·["·s·o·u·r·c·e·:·c·h·a·t·"]·,"·s·p·o·o·f·_w·a·r·n·i·n·g·":·f·a·l·s·e·,"·f·o
·l·d·e·r·":·"·i·n·b·o·x·","·b·o·d·y·":·"·o·k·","·h·t·m·l·_b·o·d·y·":·n·u·l·l·,"·s·u·b·j
·e·c·t·":·n·u·l·l·,"·h·a·s·_a·t·t·a·c·h·m·e·n·t·":·f·a·l·s·e·,"·a·t·t·a·c·h·m·e·n·t·s·":·[
]·,"·r·a·w·_a·t·t·a·c·h·m·e·n·t·s·":·n·u·l·l·,"·t·h·r·e·a·d·_i·d·":·"·i·d·:·2·4·2·2·3·7·0
·4·2·5·4·8·1·7·6·","·a·c·t·i·o·n·_i·d·":·"·1·3·5·5·0·2·2·4·7·1·6·7·3·0·0·0·0·0·0·","·
a·c·t·i·o·n·_t·y·p·e·":·"·m·a·t·t·y·p·e·:·u·s·e·r·-·g·e·n·e·r·a·t·e·d·-
·m·e·s·s·a·g·e·"}·],"·r·o·g·e·r·":·{"·i·d·:·2·4·2·2·3·7·0·4·2·5·4·8·1·7·6·":·{"·1·0·0
·0·0·3·8·6·1·2·8·4·0·6·1·":·1·3·5·5·0·2·2·4·6·6·8·2·9·}}·,"·p·a·y·l·o·a·d·_s·o·u·r·c·
e·":·"·s·e·r·v·e·r·_i·n·i·t·i·a·l·_d·a·t·a·"}

```

Appendix 26 – Scenario 2 Suspect Google+ Account Artefact



Appendix 27 - Scenario 2 Google+ Photo Upload URL History

Target	Domain	URL	Visit Count	User	LastAccessed	Browser	Job
John01	plus.google.com/	https://plus.google.com/_/upload/photos/resumable?authuser=0&upload_id=AEnB2Uo7irt8P_UgITbw4ucT6eQeaDqnCY0i4ffV-mZsxsjxket92wDAp9k0RpApQ-SYyzQdhhEzkwW_NWMzhl0NxCBMKdwBQ&file_id=000	2	machine 2	17/11/12 08:11:11p.m.	Internet Explorer (Windows)	Evidence Processor
John01	plus.google.com/	https://plus.google.com/_/upload/photos/resumable?authuser=0&upload_id=AEnB2UpCuBz_f-KGHVv5TIANzqwgVcoEOV_mxs80Jfl-e12rfBRAjs4PUrbUYD7VY0SeQkVFCiGPbC4JQ5GWZmGrqHMReN1VyQ&file_id=000	2	machine 2	17/11/12 08:38:51p.m.	Internet Explorer (Windows)	Evidence Processor
John01	plus.google.com/	https://plus.google.com/_/upload/photos/resumable?authuser=0&upload_id=AEnB2UpLPU1mRfGcDOWOyQ_Eh18uHsJ3sfKYToKhB2Q3yUws hOZHEPfNE3A9aSRdzk83agt_vCnN17SleSMBv8QC8DIQwGYUBA&file_id=000	2	machine 2	17/11/12 08:09:29p.m.	Internet Explorer (Windows)	Evidence Processor
John01	plus.google.com/	https://plus.google.com/_/upload/photos/resumable?authuser=0&upload_id=AEnB2Uq3NqWN_fzMf0uZOXsD0N5Do_jG7vbFkyAMCc_42UDaaeTF130I39rXSW9fc75jISsbAOqqGf9F-xX6MK5u3cBFtSiRgg&file_id=000	2	machine 2	17/11/12 08:10:17p.m.	Internet Explorer (Windows)	Evidence Processor
John01	plus.google.com/	https://plus.google.com/_/upload/photos/resumable?authuser=0&upload_id=AEnB2UqeLRT1K0g2BW6sASBFKoXhaI7YB-M29shcfmeUrBj629G4nAVh7LncpG2nO8L8NeubZK2_Uu4g6iGCXUbW9RNfzfYT-Q&file_id=000	2	machine 2	17/11/12 08:37:39p.m.	Internet Explorer (Windows)	Evidence Processor
John01	plus.google.com/	https://plus.google.com/_/upload/photos/resumable?authuser=0&upload_id=AEnB2UqX1LujyriUeo6-w6KmAyN76yhxjpGLTgeTd8CJT5zniGIH5Fhhr2Xs7H_-npCtzEmKqYFbzNuGgBJwPdU1M7Rc6u2Ibg&file_id=000	2	machine 2	17/11/12 08:38:17p.m.	Internet Explorer (Windows)	Evidence Processor
John01	plus.google.com/	https://plus.google.com/_/upload/photos/resumable?authuser=0&upload_id=AEnB2UrTLeWuhpGBaaRPHWoWc_gd90_4HoMBeaD5Q6t7Ir94fqVaRzTbU7dfhbGASg4IEtHcKsHNhpQkXez-s90rJrBv4yfjCg&file_id=000	2	machine 2	17/11/12 08:10:42p.m.	Internet Explorer (Windows)	Evidence Processor

Appendix 28 - Scenario 2 Suspected Images found in Browser Cache

MD5	Primary Device	Item Path	True Path	Message Size	URL Name	Internet Artifact Type	Record Last Accessed	Visit Count	File Name	Url Host
e08abda446c2967ccc6d65415fb9c773	John01	Internet Explorer (Windows)\Cache\Image\IMG_2488[2].jpg	scenario2\Cache\Image\IMG_2488[2].jpg	79109	https://lh5.googleusercontent.com/-B2GUvIG0UIY/UKc4jsUCSsI/AAAAAAAAAJw/0x7ZWTXZeaw/s449/IMG_2488.jpg	Cache\Image	17/11/12 08:19:41p.m.	1	IMG_2488[2].jpg	lh5.googleusercontent.com/
e08abda446c2967ccc6d65415fb9c773	John01	Internet Explorer (Windows)\Cache\Image\IMG_2488[1].jpg	scenario2\Cache\Image\IMG_2488[1].jpg	79109	https://lh3.googleusercontent.com/-B2GUvIG0UIY/UKc4jsUCSsI/AAAAAAAAAJw/0x7ZWTXZeaw/s449/IMG_2488.jpg	Cache\Image	17/11/12 08:19:42p.m.	1	IMG_2488[1].jpg	lh3.googleusercontent.com/
a7e92f7d030733671163606896a7913a	John01	Internet Explorer (Windows)\Cache\Image\IMG_6657[1].jpg	scenario2\Cache\Image\IMG_6657[1].jpg	126995	https://lh6.googleusercontent.com/-dULXhXnzG28/UKc4cpAz6SI/AAAAAAAAAJY/892H9qa3nw8/s599/IMG_6657.jpg	Cache\Image	17/11/12 08:20:16p.m.	1	IMG_6657[1].jpg	lh6.googleusercontent.com/
a7e92f7d030733671163606896a7913a	John01	Internet Explorer (Windows)\Cache\Image\IMG_6657[2].jpg	scenario2\Cache\Image\IMG_6657[2].jpg	126995	https://lh4.googleusercontent.com/-dULXhXnzG28/UKc4cpAz6SI/AAAAAAAAAJY/892H9qa3nw8/s599/IMG_6657.jpg	Cache\Image	17/11/12 08:20:16p.m.	1	IMG_6657[2].jpg	lh4.googleusercontent.com/
9d8a810611261944650efbaea41b8e09	John01	Internet Explorer (Windows)\Cache\Image\IMG_7431[1].jpg	scenario2\Cache\Image\IMG_7431[1].jpg	87605	https://lh4.googleusercontent.com/-Si07QCvcTco/UKc4V9ked4I/AAAAAAAAAI4/	Cache\Image	17/11/12 08:20:21p.m.	1	IMG_7431[1].jpg	lh4.googleusercontent.com/

MD5	Primary Device	Item Path	True Path	Message Size	URL Name	Internet Artifact Type	Record Last Accessed	Visit Count	File Name	Url Host
					EKA1HYVrIKQ/s599/IMG_7431.jpg					
a46d2f59c332f875745696b0c805e1a9	John01	Internet Explorer (Windows)\Cache\Image\IMG_2255[2].jpg	scenario2\Cache\Image\IMG_2255[2].jpg	50365	https://lh4.googleusercontent.com/-IIGr99YwJE/UKc4KBAvHHI/AAAAAAAAAIc/X94PGF_kDPE/s449/IMG_2255.jpg	Cache\Image	17/11/12 08:20:27p.m.	1	IMG_2255[2].jpg	lh4.googleusercontent.com/
a46d2f59c332f875745696b0c805e1a9	John01	Internet Explorer (Windows)\Cache\Image\IMG_2255[1].jpg	scenario2\Cache\Image\IMG_2255[1].jpg	50365	https://lh6.googleusercontent.com/-IIGr99YwJE/UKc4KBAvHHI/AAAAAAAAAIc/X94PGF_kDPE/s449/IMG_2255.jpg	Cache\Image	17/11/12 08:20:28p.m.	1	IMG_2255[1].jpg	lh6.googleusercontent.com/
38ace8847187b4e93478ecbedad88ce7	John01	Internet Explorer (Windows)\Cache\Image\IMG_2255[2].jpg	scenario2\Cache\Image\IMG_2255[2].jpg	26433	https://lh3.googleusercontent.com/-IIGr99YwJE/UKc4KBAvHHI/AAAAAAAAAIc/X94PGF_kDPE/w497-h373/IMG_2255.jpg	Cache\Image	17/11/12 08:27:08p.m.	3	IMG_2255[2].jpg	lh3.googleusercontent.com/
ad3ecd3fd7a7cb2535d5ce903921eb9a	John01	Internet Explorer (Windows)\Cache\Image\IMG_2488[3].jpg	scenario2\Cache\Image\IMG_2488[3].jpg	43051	https://lh4.googleusercontent.com/-B2GUvIG0UIY/UKc4jsUCSsI/AAAAAAAAAJw/0x7ZWTXZeaw/w497-h373/IMG_2488.jpg	Cache\Image	17/11/12 08:27:08p.m.	3	IMG_2488[3].jpg	lh4.googleusercontent.com/

MD5	Primary Device	Item Path	True Path	Message Size	URL Name	Internet Artifact Type	Record Last Accessed	Visit Count	File Name	Url Host
7257c41f1d08533e0dc717cb75b86be8	John01	Internet Explorer (Windows)\Cache\Image\IMG_6657[1].jpg	scenario2\Cache\Image\IMG_6657[1].jpg	80510	https://lh5.googleusercontent.com/-dULXhXnzG28/UKc4cpAz6SI/AAAAAAAAAJY/892H9qa3nw8/w497-h373/IMG_6657.jpg	Cache\Image	17/11/12 08:27:08p.m.	3	IMG_6657[1].jpg	lh5.googleusercontent.com/
0272e9e86d438cf71e778ea27c11eba3	John01	Internet Explorer (Windows)\Cache\Image\IMG_7431[2].jpg	scenario2\Cache\Image\IMG_7431[2].jpg	53805	https://lh5.googleusercontent.com/-Si07QCvcTco/UKc4V9ked4I/AAAAAAAAAI4/EKA1HYVrIKQ/w497-h373/IMG_7431.jpg	Cache\Image	17/11/12 08:27:08p.m.	3	IMG_7431[2].jpg	lh5.googleusercontent.com/
67767de17416f20482dc2c3edca25b6b	John01	Internet Explorer (Windows)\Cache\Image\IMG_6677[1].jpg	scenario2\Cache\Image\IMG_6677[1].jpg	36909	https://lh3.googleusercontent.com/-IiLoZI47Gio/UKc-wm-BBSI/AAAAAAAAAKw/yLkIR54p_rU/w497-h373/IMG_6677.jpg	Cache\Image	17/11/12 08:37:41p.m.	1	IMG_6677[1].jpg	lh3.googleusercontent.com/
7bf281181201be09076c223ccfe2d0df	John01	Internet Explorer (Windows)\Cache\Image\IMG_8434[1].jpg	scenario2\Cache\Image\IMG_8434[1].jpg	50934	https://lh4.googleusercontent.com/-1BQNEyP-DPg/UKc-6EXw92I/AAAAAAAAALM/W7kvH1DCdG4/w497-h373/IMG_8434.jpg	Cache\Image	17/11/12 08:38:19p.m.	1	IMG_8434[1].jpg	lh4.googleusercontent.com/
c49a7847741c2396e218861060986f9d	John01	Internet Explorer (Windows)\Cache\Image\IMG_2292[1].jpg	scenario2\Cache\Image\IMG_2292[1].jpg	52990	https://lh5.googleusercontent.com/-KN4ArRUo-ZQ/UKc_CqSjOXI/AAAAAAAAALo/3shHb4zNT_Q/w497-	Cache\Image	17/11/12 08:38:53p.m.	1	IMG_2292[1].jpg	lh5.googleusercontent.com/

MD5	Primary Device	Item Path	True Path	Message Size	URL Name	Internet Artifact Type	Record Last Accessed	Visit Count	File Name	Url Host
					h373/IMG_2292.jpg					
eef346cadfecac208d1d8623b0516687	John01	Internet Explorer (Windows)\Cache\Image\IMG_2292[1].jpg	scenario2\Cache\Image\IMG_2292[1].jpg	84801	https://lh6.googleusercontent.com/-KN4ArRUo-ZQ/UKc-CqSjOXI/AAAAAAAAALo/3shHb4zNT_Q/s599/IMG_2292.jpg	Cache\Image	17/11/12 08:39:25p.m.	1	IMG_2292[1].jpg	lh6.googleusercontent.com/
eef346cadfecac208d1d8623b0516687	John01	Internet Explorer (Windows)\Cache\Image\IMG_2292[2].jpg	scenario2\Cache\Image\IMG_2292[2].jpg	84801	https://lh4.googleusercontent.com/-KN4ArRUo-ZQ/UKc-CqSjOXI/AAAAAAAAALo/3shHb4zNT_Q/s599/IMG_2292.jpg	Cache\Image	17/11/12 08:39:26p.m.	1	IMG_2292[2].jpg	lh4.googleusercontent.com/
d63b204e937d6ff9b560861ec40ff9b1	John01	Internet Explorer (Windows)\Cache\Image\IMG_8434[1].jpg	scenario2\Cache\Image\IMG_8434[1].jpg	98346	https://lh5.googleusercontent.com/-1BQNEyP-DPg/UKc-6EXw92I/AAAAAAAAALM/W7kvH1DCdG4/s449/IMG_8434.jpg	Cache\Image	17/11/12 08:39:43p.m.	1	IMG_8434[1].jpg	lh5.googleusercontent.com/
d63b204e937d6ff9b560861ec40ff9b1	John01	Internet Explorer (Windows)\Cache\Image\IMG_8434[2].jpg	scenario2\Cache\Image\IMG_8434[2].jpg	98346	https://lh3.googleusercontent.com/-1BQNEyP-DPg/UKc-6EXw92I/AAAAAAAAALM/W7kvH1DCdG4/s449/IMG_8434.jpg	Cache\Image	17/11/12 08:39:44p.m.	1	IMG_8434[2].jpg	lh3.googleusercontent.com/

MD5	Primary Device	Item Path	True Path	Message Size	URL Name	Internet Artifact Type	Record Last Accessed	Visit Count	File Name	Url Host
b4f2f51afaed3c46f073d3266ccc7782	John01	Internet Explorer (Windows)\Cache\Image\IMG_6677[1].jpg	scenario2\Cache\Image\IMG_6677[1].jpg	68634	https://lh6.googleusercontent.com/-IiLoZI47Gio/UKc-wm-BBSI/AAAAAAAAAKw/yLkIR54p_rU/s449/IMG_6677.jpg	Cache\Image	17/11/12 08:39:51p.m.	1	IMG_6677[1].jpg	lh6.googleusercontent.com/
fc7417494f254cb6f9221d872f478f55	John01	Internet Explorer (Windows)\Cache\Image\IMG_2292[1].jpg	scenario2\Cache\Image\IMG_2292[1].jpg	16590	https://lh6.googleusercontent.com/-wJp-9UkW4NA/UKc-Cr19zwE/AAAAAAAAALo/1JM1aS4z3lE/s180-c/2012111607	Cache\Image	17/11/12 08:40:04p.m.	1	IMG_2292[1].jpg	lh6.googleusercontent.com/
3775052a71412842ba0f3981b109966e	John01	Internet Explorer (Windows)\Cache\Image\IMG_6677[2].jpg	scenario2\Cache\Image\IMG_6677[2].jpg	18240	https://lh3.googleusercontent.com/-IiLoZI47Gio/UKc-wm-BBSI/AAAAAAAAAKw/yLkIR54p_rU/s180-c/photo.jpg	Cache\Image	17/11/12 08:40:04p.m.	1	IMG_6677[2].jpg	lh3.googleusercontent.com/
a659f42c159e18adc4889027c10a9ccd	John01	Internet Explorer (Windows)\Cache\Image\IMG_8434[2].jpg	scenario2\Cache\Image\IMG_8434[2].jpg	24807	https://lh4.googleusercontent.com/-1BQNEyP-DPg/UKc-6EXw92I/AAAAAAAAALM/W7kvH1DCdG4/s180-c/photo.jpg	Cache\Image	17/11/12 08:40:04p.m.	1	IMG_8434[2].jpg	lh4.googleusercontent.com/
f1ff5591a24b4f7bef198c786efe2d2c	John01	Internet Explorer (Windows)\Cache\Image\IMG_2292[2].jpg	scenario2\Cache\Image\IMG_2292[2].jpg	33368	https://lh6.googleusercontent.com/-wJp-9UkW4NA/UKc-Cr19zwE/AAAAAAAAALo/1JM1aS4z3lE/s297-c/2012111607	Cache\Image	17/11/12 08:40:45p.m.	1	IMG_2292[2].jpg	lh6.googleusercontent.com/

MD5	Primary Device	Item Path	True Path	Message Size	URL Name	Internet Artifact Type	Record Last Accessed	Visit Count	File Name	Url Host
a0357137732fb0ab80bbcd7fc3bcd7b5	John01	Internet Explorer (Windows)\Cache\Image\IMG_6677[1].jpg	scenario2\Cache\Image\IMG_6677[1].jpg	41715	https://lh3.googleusercontent.com/-IiLoZI47Gio/UKc-wm-BBSI/AAAAAAAAAKw/yLkIR54p_rU/s297-c/photo.jpg	Cache\Image	17/11/12 08:40:45p.m.	1	IMG_6677[1].jpg	lh3.googleusercontent.com/
804dd916a1fcc7541ef7feae5a4c9d94	John01	Internet Explorer (Windows)\Cache\Image\IMG_8434[3].jpg	scenario2\Cache\Image\IMG_8434[3].jpg	57897	https://lh4.googleusercontent.com/-1BQNEyP-DPg/UKc-6EXw92I/AAAAAAAAALM/W7kvH1DCdG4/s297-c/photo.jpg	Cache\Image	17/11/12 08:40:45p.m.	1	IMG_8434[3].jpg	lh4.googleusercontent.com/
db159e8e68a6639646460581bb24e917	John01	Internet Explorer (Windows)\Cache\Image\IMG_2255[1].jpg	scenario2\Cache\Image\IMG_2255[1].jpg	27313	https://lh5.googleusercontent.com/-EkAfDRePUlo/UKc4J1iPTcE/AAAAAAAAAIc/67T1jOVVzhI/s297-c/20121116	Cache\Image	17/11/12 08:40:46p.m.	1	IMG_2255[1].jpg	lh5.googleusercontent.com/
e2307772c49ae397462c9d5df14ed47e	John01	Internet Explorer (Windows)\Cache\Image\IMG_2488[1].jpg	scenario2\Cache\Image\IMG_2488[1].jpg	46716	https://lh6.googleusercontent.com/-1tO7zOw3Nds/UKc4jnyR7yE/AAAAAAAAAJw/uRG89_UStN0/s297-c/2012111604	Cache\Image	17/11/12 08:40:46p.m.	1	IMG_2488[1].jpg	lh6.googleusercontent.com/
3dfc6cd96430e5cb2eae5b5da96cc9dc	John01	Internet Explorer (Windows)\Cache\Image\IMG_6657[1].jpg	scenario2\Cache\Image\IMG_6657[1].jpg	47567	https://lh4.googleusercontent.com/-cYN9LWIa_po/UKc4cSS0FjE/AAAAAAAAAJY/vKZDbf5vZMI/s297-c/2012111603	Cache\Image	17/11/12 08:40:46p.m.	1	IMG_6657[1].jpg	lh4.googleusercontent.com/

MD5	Primary Device	Item Path	True Path	Message Size	URL Name	Internet Artifact Type	Record Last Accessed	Visit Count	File Name	Url Host
a0357137732fb0ab80bbcd7fc3bcd7b5	John01	Internet Explorer (Windows)\Cache\Image\IMG_6677[2].jpg	scenario2\Cache\Image\IMG_6677[2].jpg	41715	https://lh5.googleusercontent.com/-bpBIebUthms/UKc-wnYAEQE/AAAAAAAAAAKw/4jGc8cdZde4/s297-c/2012111605	Cache\Image	17/11/12 08:40:46p.m.	1	IMG_6677[2].jpg	lh5.googleusercontent.com/
804dd916a1fcc7541ef7feae5a4c9d94	John01	Internet Explorer (Windows)\Cache\Image\IMG_8434[1].jpg	scenario2\Cache\Image\IMG_8434[1].jpg	57897	https://lh5.googleusercontent.com/-NesTq19qJ4k/UKc-5y8TR1E/AAAAAAAAALM/tgMphcraV_4/s297-c/2012111606	Cache\Image	17/11/12 08:40:46p.m.	1	IMG_8434[1].jpg	lh5.googleusercontent.com/
a232795eba4400ffeb41f88128dcd4e1	John01	Internet Explorer (Windows)\Cache\Image\IMG_7431[2].jpg	scenario2\Cache\Image\IMG_7431[2].jpg	37742	https://lh5.googleusercontent.com/-pL4NFYIUks/UKc4V0qr7gE/AAAAAAAAAI4/bKBRydN7jKE/s297-c/2012111602	Cache\Image	17/11/12 08:40:47p.m.	1	IMG_7431[2].jpg	lh5.googleusercontent.com/

Appendix 29 – Scenario 2 Suspected Steganographic Images in Suspect’s Hard Drive

No	Name	File Ext	Logical Size	File Type	File Created	MD5	Item Path	Physical Size	Evidence File	Identified as Steganographic Images	Secret Message Extracted
1	IMG_2255.jpg	jpg	92,879	JPEG Image Standard	17/11/12 08:00:50p.m.	f4e533ad140befd2f05c58fb645e979a	John01\D\Users\machine2\Pictures\IMG_2255.jpg	94,208	John01	Suspected	No
2	IMG_7431.jpg	jpg	115,913	JPEG Image Standard	17/11/12 08:03:07p.m.	ebdef169ed39cea0becac80a6acfe4f3	John01\D\Users\machine2\Pictures\IMG_7431.jpg	118,784	John01	Suspected	No
3	IMG_2488.jpg	jpg	158,836	JPEG Image Standard	17/11/12 08:05:19p.m.	74d4f86fe44f3b95d2e82fcba6919559	John01\D\Users\machine2\Pictures\IMG_2488.jpg	159,744	John01	Suspected	No
4	IMG_6657.jpg	jpg	172,180	JPEG Image Standard	17/11/12 08:06:52p.m.	ba2b654bb65acff79e444737c6c71c277	John01\D\Users\machine2\Pictures\IMG_6657.jpg	176,128	John01	Suspected	No
5	IMG_6677.jpg	jpg	143,530	JPEG Image Standard	17/11/12 08:28:54p.m.	16598c670f034587ac4a26c67d533be7	John01\D\Users\machine2\Pictures\IMG_6677.jpg	147,456	John01	Suspected	No
6	IMG_8434.jpg	jpg	192,200	JPEG Image Standard	17/11/12 08:30:34p.m.	85b48065d865d8f6f97b2cd46f15409f	John01\D\Users\machine2\Pictures\IMG_8434.jpg	192,512	John01	Suspected	No
7	IMG_2292.jpg	jpg	113,195	JPEG Image Standard	17/11/12 08:33:32p.m.	8bc42cfba96581913d6cb1a96c9385e2	John01\D\Users\machine2\Pictures\IMG_2292.jpg	114,688	John01	Suspected	No

Appendix 30 – Scenario 2 Suspicious File Activities

File Type	MD5	True Path	Profile Name	Url Name	Type	Record Last Accessed	Visit Count	Internet Artifact Type	Browser Type
IE Cache Index dat	066c5fade9bd92c5c7596d9bf2c23e17	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20Doe/photos/IMG_2255.JPG	URL	17/11/12 08:00:04p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	066c5fade9bd92c5c7596d9bf2c23e17	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20Doe/photos/IMG_2255.JPG	URL	17/11/12 08:00:04p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	066c5fade9bd92c5c7596d9bf2c23e17	scenario2\History\Visited Link\index.dat	machine2	file:///E:/John%20Doe/photos/IMG_2255.JPG	URL	17/11/12 08:00:04p.m.	1	History\ Visited Link	Internet Explorer (Windows)
	592da44298b4f8de3856f68ef02e388e	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/promo25_11.txt	URL	17/11/12 08:00:19p.m.	1	History\ Visited Link	Internet Explorer (Windows)
	592da44298b4f8de3856f68ef02e388e	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/promo25_11.txt	URL	17/11/12 08:00:19p.m.	1	History\ Visited Link	Internet Explorer (Windows)
Data ASCII & Binary	592da44298b4f8de3856f68ef02e388e	scenario2\History\Visited Link\index.dat	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/promo25_11.txt	URL	17/11/12 08:00:19p.m.	1	History\ Visited Link	Internet Explorer (Windows)

File Type	MD5	True Path	Profile Name	Url Name	Type	Record Last Accessed	Visit Count	Internet Artifact Type	Browser Type
IE Cache Index dat	81282810d96a6e5cfa042e7652b8c7df	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/machine2/Pictures/IMG_2255.jpg	URL	17/11/12 08:00:50p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	7807a637eae1841488da2fd9b6bdf1a	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20Doe/photos/IMG_7431.JPG	URL	17/11/12 08:01:30p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	7807a637eae1841488da2fd9b6bdf1a	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20Doe/photos/IMG_7431.JPG	URL	17/11/12 08:01:30p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	7807a637eae1841488da2fd9b6bdf1a	scenario2\History\Visited Link\index.dat	machine2	file:///E:/John%20Doe/photos/IMG_7431.JPG	URL	17/11/12 08:01:30p.m.	1	History\ Visited Link	Internet Explorer (Windows)
	b1c597c2904a3e5b067d86ed8868c960	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/sales%20strategy.txt	URL	17/11/12 08:01:54p.m.	1	History\ Visited Link	Internet Explorer (Windows)
	b1c597c2904a3e5b067d86ed8868c960	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/sales%20strategy.txt	URL	17/11/12 08:01:54p.m.	1	History\ Visited Link	Internet Explorer (Windows)

File Type	MD5	True Path	Profile Name	Url Name	Type	Record Last Accessed	Visit Count	Internet Artifact Type	Browser Type
Data ASCII & Binary	b1c597c2904a3e5b067d86ed8868c960	scenario2\History\Visited Link\index.dat	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/sales%20strategy.txt	URL	17/11/12 08:01:54p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	d6e4ad4df63ef91bd1725b251a98e5bc	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/machine2/Pictures/IMG_7431.jpg	URL	17/11/12 08:02:33p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	cf366a6c968aa1ee6d281957a885bc7d	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20Doe/photos/IMG_2488.JPG	URL	17/11/12 08:04:26p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	cf366a6c968aa1ee6d281957a885bc7d	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20Doe/photos/IMG_2488.JPG	URL	17/11/12 08:04:26p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	cf366a6c968aa1ee6d281957a885bc7d	scenario2\History\Visited Link\index.dat	machine2	file:///E:/John%20Doe/photos/IMG_2488.JPG	URL	17/11/12 08:04:26p.m.	1	History\ Visited Link	Internet Explorer (Windows)
	be7109b2b18aad1d7e332499dfa3f603	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/market%20Analysis.txt	URL	17/11/12 08:04:44p.m.	1	History\ Visited Link	Internet Explorer (Windows)

File Type	MD5	True Path	Profile Name	Url Name	Type	Record Last Accessed	Visit Count	Internet Artifact Type	Browser Type
IE Cache Index dat	ab44da68ed29c8f5decd793297c219c0	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/mac hine2/Pictures/IMG_2488.jpg	URL	17/11/12 08:05:20p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	ab44da68ed29c8f5decd793297c219c0	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/mac hine2/Pictures/IMG_2488.jpg	URL	17/11/12 08:05:20p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	ab44da68ed29c8f5decd793297c219c0	scenario2\History\Visited Link\index.dat	machine2	file:///C:/Users/mac hine2/Pictures/IMG_2488.jpg	URL	17/11/12 08:05:20p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	cffd517ac48e9c589a094394a6076a1f	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20D oe/photos/IMG_6657.JPG	URL	17/11/12 08:06:11p.m.	2	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	cffd517ac48e9c589a094394a6076a1f	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20D oe/photos/IMG_6657.JPG	URL	17/11/12 08:06:11p.m.	2	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	cffd517ac48e9c589a094394a6076a1f	scenario2\History\Visited Link\index.dat	machine2	file:///E:/John%20D oe/photos/IMG_6657.JPG	URL	17/11/12 08:06:11p.m.	2	History\ Visited Link	Internet Explorer (Windows)

File Type	MD5	True Path	Profile Name	Url Name	Type	Record Last Accessed	Visit Count	Internet Artifact Type	Browser Type
	293b023b78770df54c63d58ba700d95e	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/use%20of%20fun ds.txt	URL	17/11/12 08:06:37p.m.	2	History\ Visited Link	Internet Explorer (Windows)
Data ASCII & Binary	293b023b78770df54c63d58ba700d95e	scenario2\History\Visited Link\index.dat	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/use%20of%20fun ds.txt	URL	17/11/12 08:06:37p.m.	2	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	207d8da3633d352e1fc7231391da2c04	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/mac hine2/Pictures/IMG _6657.jpg	URL	17/11/12 08:07:57p.m.	3	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	207d8da3633d352e1fc7231391da2c04	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/mac hine2/Pictures/IMG _6657.jpg	URL	17/11/12 08:07:57p.m.	3	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	207d8da3633d352e1fc7231391da2c04	scenario2\History\Visited Link\index.dat	machine2	file:///C:/Users/mac hine2/Pictures/IMG _6657.jpg	URL	17/11/12 08:07:57p.m.	3	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	f0cf2d0d90f124dc63bb76e3d7170aae	scenario2\History\Visited Link\{24615c34-3084-11e2-b7d1-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/mac hine2/Desktop/note. txt	URL	17/11/12 08:08:08p.m.	1	History\ Visited Link	Internet Explorer (Windows)

File Type	MD5	True Path	Profile Name	Url Name	Type	Record Last Accessed	Visit Count	Internet Artifact Type	Browser Type
IE Cache Index dat	a1cc36936c7a0c28d36a2f3103b581ab	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20Doe/photos/IMG_6677.JPG	URL	17/11/12 08:28:07p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	a1cc36936c7a0c28d36a2f3103b581ab	scenario2\History\Visited Link\index.dat	machine2	file:///E:/John%20Doe/photos/IMG_6677.JPG	URL	17/11/12 08:28:07p.m.	1	History\ Visited Link	Internet Explorer (Windows)
Data ASCII & Binary	9c0512ad9065ee9be6adc2496b bc321b	scenario2\History\Visited Link\index.dat	machine2	file:///C:/StarWorld %20Sales%20&%20Marketing%20Dep t/promo02_12.txt	URL	17/11/12 08:28:31p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	4016138a6a0be3cd6bb9e28af63a35d4	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20Doe/photos/IMG_8434.JPG	URL	17/11/12 08:29:29p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	4016138a6a0be3cd6bb9e28af63a35d4	scenario2\History\Visited Link\index.dat	machine2	file:///E:/John%20Doe/photos/IMG_8434.JPG	URL	17/11/12 08:29:29p.m.	1	History\ Visited Link	Internet Explorer (Windows)
	9c393451cef9fc109c259689b2a46902	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}\{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/StarWorld %20Sales%20&%20Marketing%20Dep t/market%20Analys is.txt	URL	17/11/12 08:29:53p.m.	2	History\ Visited Link	Internet Explorer (Windows)
Data ASCII & Binary	9c393451cef9fc109c259689b2a46902	scenario2\History\Visited Link\index.dat	machine2	file:///C:/StarWorld %20Sales%20&%20Marketing%20Dep t/market%20Analys	URL	17/11/12 08:29:53p.m.	2	History\ Visited Link	Internet Explorer (Windows)

File Type	MD5	True Path	Profile Name	Url Name	Type	Record Last Accessed	Visit Count	Internet Artifact Type	Browser Type
				is.txt					
IE Cache Index dat	e7502c235f2b66cc63b37587d42b21ec	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20Doe/photos/IMG_6677.JPG	URL	17/11/12 08:30:51p.m.	2	History\Visited Link	Internet Explorer (Windows)
IE Cache Index dat	e7502c235f2b66cc63b37587d42b21ec	scenario2\History\Visited Link\index.dat	machine2	file:///E:/John%20Doe/photos/IMG_6677.JPG	URL	17/11/12 08:30:51p.m.	2	History\Visited Link	Internet Explorer (Windows)
	8f1f7af67a7cd1bbcc5acc899721a178	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/promo02_12.txt	URL	17/11/12 08:31:02p.m.	2	History\Visited Link	Internet Explorer (Windows)
Data ASCII & Binary	8f1f7af67a7cd1bbcc5acc899721a178	scenario2\History\Visited Link\index.dat	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/promo02_12.txt	URL	17/11/12 08:31:02p.m.	2	History\Visited Link	Internet Explorer (Windows)
IE Cache Index dat	3ba55dd00990c18ecac83610ceb45321	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/machine2/Pictures/IMG_6677.jpg	URL	17/11/12 08:31:19p.m.	2	History\Visited Link	Internet Explorer (Windows)
IE Cache Index dat	3ba55dd00990c18ecac83610ceb45321	scenario2\History\Visited Link\index.dat	machine2	file:///C:/Users/machine2/Pictures/IMG_6677.jpg	URL	17/11/12 08:31:19p.m.	2	History\Visited Link	Internet Explorer (Windows)

File Type	MD5	True Path	Profile Name	Url Name	Type	Record Last Accessed	Visit Count	Internet Artifact Type	Browser Type
IE Cache Index dat	de15d6147865f5a615ce69baedc9c120	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///E:/John%20Doe/photos/IMG_2292.JPG	URL	17/11/12 08:33:01p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	de15d6147865f5a615ce69baedc9c120	scenario2\History\Visited Link\index.dat	machine2	file:///E:/John%20Doe/photos/IMG_2292.JPG	URL	17/11/12 08:33:01p.m.	1	History\ Visited Link	Internet Explorer (Windows)
Data ASCII & Binary	ba5e1656beb096ebe2b98f46ade0b5e	scenario2\History\Visited Link\index.dat	machine2	file:///C:/StarWorld%20Sales%20&%20Marketing%20Dept/cash%20flow%20projection.txt	URL	17/11/12 08:33:17p.m.	1	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	330a42809abe8edc83f7f8a2841a0b7d	scenario2\History\Visited Link\index.dat	machine2	file:///C:/Users/machine2/Pictures/IMG_2255.jpg	URL	17/11/12 08:33:53p.m.	2	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	b6902454e8235e253e2c26e37a46898d	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/machine2/Desktop/note.txt	URL	17/11/12 08:34:31p.m.	3	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	b6902454e8235e253e2c26e37a46898d	scenario2\History\Visited Link\index.dat	machine2	file:///C:/Users/machine2/Desktop/note.txt	URL	17/11/12 08:34:31p.m.	3	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	4d186005c8675e9b6bf146916bc131d7	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/machine2/Pictures/IMG_7431.jpg	URL	17/11/12 08:34:50p.m.	2	History\ Visited Link	Internet Explorer (Windows)

File Type	MD5	True Path	Profile Name	Url Name	Type	Record Last Accessed	Visit Count	Internet Artifact Type	Browser Type
IE Cache Index dat	4d186005c8675e9b6bf146916bc131d7	scenario2\History\Visited Link\index.dat	machine2	file:///C:/Users/machine2/Pictures/IMG_7431.jpg	URL	17/11/12 08:34:50p.m.	2	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	73cdbf4508e337c4db352f4ea2c1454b	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/machine2/Pictures/IMG_8434.jpg	URL	17/11/12 08:35:40p.m.	2	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	73cdbf4508e337c4db352f4ea2c1454b	scenario2\History\Visited Link\index.dat	machine2	file:///C:/Users/machine2/Pictures/IMG_8434.jpg	URL	17/11/12 08:35:40p.m.	2	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	4587135446a48ab60d79527256523374	scenario2\History\Visited Link\{eb28885c-3087-11e2-b63e-00266c4990d3}{3808876b-c176-4e48-b7ae-04046e6cc752}	machine2	file:///C:/Users/machine2/Pictures/IMG_2292.jpg	URL	17/11/12 08:36:27p.m.	2	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	4587135446a48ab60d79527256523374	scenario2\History\Visited Link\index.dat	machine2	file:///C:/Users/machine2/Pictures/IMG_2292.jpg	URL	17/11/12 08:36:27p.m.	2	History\ Visited Link	Internet Explorer (Windows)
IE Cache Index dat	de4f3d246bb33f49dd1c471d2b2eb03a	scenario2\History\Visited Link\index.dat	machine2	file:///C:/Users/machine2/Desktop/note 2.txt	URL	17/11/12 08:36:35p.m.	1	History\ Visited Link	Internet Explorer (Windows)


Appendix 31 – Scenario 2 Google+ Message Posted (Keyword Search)

Suspicious content in google+	Converted to Local Time	Found in
["up",""," Google+ "," John Doe ","you deserve this! haha.. ", 1353136285580 ,"http://www.google.com/favicon.ico",[], ,"z12gevwbmisiyydphi04cevghovq3vpm4xwk0k","","s: updates:esshare ",[[,,,,"",[,," https://lh4.googleusercontent.com/- B2GUvIG0UIY/UKc4jsUCSsI/AAAAAAAAAJw/0x7ZWTXZeaw/IMG_2488.jpg ",640,480]	Sat, 17 Nov 2012 20:11:25 +13:00	John01\D\Windows\SoftwareDistribution\Download\593730 d50670906ac7a22ad394c1830b\package_47_for_kb2656372 ~31bf3856ad364e35~x86~~6.1.2.0.cat
["up",""," Google+ "," John Doe ","when are you free for a coffee? ", 1353136318500 ,"http://www.google.com/favicon.ico",	Sat, 17 Nov 2012 20:11:58 +13:00	John01\D\Windows\SoftwareDistribution\Download\593730 d50670906ac7a22ad394c1830b\package_47_for_kb2656372 _bf~31bf3856ad364e35~x86~~6.1.2.0.cat
[," Christian Riley ",,"c u! remember to bring the tool to show me! \uffff", 1353136421620 ,"z13thztr3qqwht5tb22qv3haxkyxglbj404#135 3136421620739",,"103817061956537937504",,"z13thztr3qqwht5tb22qv3h axkyxglbj404",0,1,"./103817061956537937504",1,,,0,[,,,,,,,,,,,,,[] ,,,,,[],0]	Sat, 17 Nov 2012 20:13:41 +13:00	John01\D\Windows\SoftwareDistribution\Download\593730 d50670906ac7a22ad394c1830b\package_47_for_kb2656372 _bf~31bf3856ad364e35~x86~~6.1.2.0.cat
[,"https://lh4.googleusercontent.com/- tl8XZIAFWRE/AAAAAAAAAAI/AAAAAAAAAAA/FYO- XrTXKwY/photo.jpg" ,,,,,,0] ,[, " John Doe ",," okie dokie! \uffff", 1353136658555 ,"z13thztr3qqwht5tb22qv3haxkyxglbj404#1 353136658555459",,"okie dokie!",	Sat, 17 Nov 2012 20:17:38 +13:00	John01\D\Windows\SoftwareDistribution\Download\593730 d50670906ac7a22ad394c1830b\package_47_for_kb2656372 _bf~31bf3856ad364e35~x86~~6.1.2.0.cat

Suspicious content in google+	Converted to Local Time	Found in
<p>on the way back home,[], "111267948980380534594",[], ",https://lh5.googleusercontent.com/- tUDtph2hzHE/AAAAAAAAAAI/AAAAAAAAABI/ufURPbuY7eU/phot o.jpg",,"on the way back home","111267948980380534594/posts/FAXpfji78oJ",135313625587999 9,0.0,"./111267948980380534594",[], ,,,,"0,1353136253858004,1,0,,0,1,"5811675902743156273",0,135313625 3858,,,1,,,0,,,[], ,,,1,1,0,,1,,,0,,5,,,[3,,,,,[["https://plus.google.com/photos/111267948980 380534594/albums/5811675902743156273/5811675908841662754","ima ge/jpeg",,"/lh5.googleusercontent.com/- dULXhXnzG28/UKc4cpAz6SI/AAAAAAAAAJY/892H9qa3nw8/h371/ IMG_6657.jpg",494,371,,,,"https://lh5.googleusercontent.com/- dULXhXnzG28/UKc4cpAz6SI/AAAAAAAAAJY/892H9qa3nw8/IMG_6 657.jpg",640,480,,,,"picasa",1, ",https://lh5.googleusercontent.com/- tUDtph2hzHE/AAAAAAAAAAI/AAAAAAAAABI/ufURPbuY7eU/phot o.jpg",,"Hi christian welcome to our club",,"111267948980380534594/posts/CAFnQm2TXSz",1353134507086 999,0.0,"./111267948980380534594",</p>	<p>Sat, 17 Nov 2012 20:10:53 +13:00</p>	<p>John01\D\Windows\SoftwareDistribution\Download\593730 d50670906ac7a22ad394c1830b\package_47_for_kb2656372 ~31bf3856ad364e35~x86~~6.1.2.0.mum</p>
<p>,https://lh5.googleusercontent.com/- tUDtph2hzHE/AAAAAAAAAAI/AAAAAAAAABI/ufURPbuY7eU/phot o.jpg",,"Hi christian welcome to our club",,"111267948980380534594/posts/CAFnQm2TXSz",1353134507086 999,0.0,"./111267948980380534594",</p>	<p>Sat, 17 Nov 2012 19:41:47 +13:00</p>	<p>John01\D\Windows\SoftwareDistribution\Download\593730 d50670906ac7a22ad394c1830b\update-bf.mum</p>

Appendix 32 – Scenario 2 Significant Registry Artefacts Identified by StegAllyzerAS on a portable StegHide application

Evidence Log: Steghide v0.5.1



Statistics

File Artifacts

Total File Artifacts Detected: 22/23

Percentage Detected: 95.7 %

False Positive/ Common File Artifacts Detected: 13/13

Unique File Artifacts Detected: 9/10

Registry Artifacts

Registry Artifacts Detected: 4/12

Percentage Detected: 33.3 %

Known File Artifacts

File	False Positive/Common	Status	Path	Time Stamp
steghide.mo	False Positive	FOUND	Multiple Detections (Double-Click for more information)	Multiple Detections (Double-Click for more information)
steghide.mo	False Positive	FOUND	Multiple Detections (Double-Click for more information)	Multiple Detections (Double-Click for more information)
steghide.mo	False Positive	FOUND	Multiple Detections (Double-Click for more information)	Multiple Detections (Double-Click for more information)
about-nls.txt	False Positive	FOUND	Multiple Detections (Double-Click for more information)	Multiple Detections (Double-Click for more information)
bugs.txt	False Positive	FOUND	Multiple Detections (Double-Click for more information)	Multiple Detections (Double-Click for more information)
copying.txt	False Positive	FOUND	Multiple Detections (Double-Click for more information)	Multiple Detections (Double-Click for more information)
credits.txt	Unique	FOUND	Multiple Detections (Double-Click for more information)	Multiple Detections (Double-Click for more information)
cygiconv-2.dll	Unique	FOUND	Multiple Detections (Double-Click for more information)	Multiple Detections (Double-Click for more information)

CRC32: 486ACAC9

MD5: E81F5FD934B5460CEF8DECAA9433EAC4

SHA1: DCC8794C518EDF2E6E26CC62DF1070A74A87A19F

SHA224: 084DB9C39F90420DC7D3EF9242C51985E99B80363545497F7C60C3AA

SHA256: C2B49DE67824E95BA881D5422BE65EB043B834DA1B7D0C4D57D76C2C6CD83D00

SHA384: 2A3C5C95D21F8C44545965DDE79B9B4E2C0CECC1452FAB7B5896D7DD674D6F53C3F9C734AE49C8AD2814BB119EC3896E

SHA512: 2876D41B45EBAE46428455E8A34E56AB31E698675CE0733ABF7B097DC180028F30334C26DAC32B396843B0DE3C23320F394A79B354B5FD02191B1A46FE0

Size: 30267

Known Registry Artifacts

Registry Key Name	Status	Hive File	Time Stamp
HKEY_CURRENT_USER\Software\Cygnus Solutions	FOUND	I:\Users\machine2\NTUSER.DAT	6/12/2012 7:53:16 p
HKEY_CURRENT_USER\Software\Cygnus Solutions\Cygwin	FOUND	I:\Users\machine2\NTUSER.DAT	6/12/2012 7:53:16 p
HKEY_CURRENT_USER\Software\Cygnus Solutions\Cygwin\mounts v2	FOUND	I:\Users\machine2\NTUSER.DAT	6/12/2012 7:53:16 p
HKEY_CURRENT_USER\Software\Cygnus Solutions\Cygwin\Program Options	FOUND	I:\Users\machine2\NTUSER.DAT	6/12/2012 7:53:16 p
HKEY_LOCAL_MACHINE\SOFTWARE\Cygnus Solutions	NOT FOUND	N/A	N/A
HKEY_LOCAL_MACHINE\SOFTWARE\Cygnus Solutions\Cygwin	NOT FOUND	N/A	N/A
HKEY_LOCAL_MACHINE\SOFTWARE\Cygnus Solutions\Cygwin\mounts v2	NOT FOUND	N/A	N/A
HKEY_LOCAL_MACHINE\SOFTWARE\Cygnus Solutions\Cygwin\Program Options	NOT FOUND	N/A	N/A
HKEY_USERS\S-1-5-21-448539723-1580436667-1202660629-500\Software\Cygnus Solutions	NOT FOUND	N/A	N/A
HKEY_USERS\S-1-5-21-448539723-1580436667-1202660629-500\Software\Cygnus Solutions\Cygwin	NOT FOUND	N/A	N/A
HKEY_USERS\S-1-5-21-448539723-1580436667-1202660629-500\Software\Cygnus Solutions\Cygwin\mounts v2	NOT FOUND	N/A	N/A
HKEY_USERS\S-1-5-21-448539723-1580436667-1202660629-500\Software\Cygnus Solutions\Cygwin\Program Options	NOT FOUND	N/A	N/A