Malware architects and rational choice: A case study analysis of three cyber-offenders.

By Sarah Morrison

44728859

Macquarie University

Faculty of Arts

Security Studies and Criminology

Master of Research – Final Paper

Submission Date: 9th October, 2017

Thank you to Dr. James Martin, Macquarie University, for his supervision and support.

Abstract

This thesis explores the cybercrime of malware through case study research. In doing so, the paper will apply rational choice theory, specifically Cornish and Clarkes (1986) rational choice perspectives, to malware architects Robert Tappan Morris and David Lee Smith, as well as alleged malware architect Evgeniy Mikhailovich Bogachev. As will be demonstrated, each of the three-case examples, spread over four decades, have contributed to the development of future malware through their architecture and design, providing a critical timeline of events and motivation.

The primary aim of the research is to establish whether rational choice is an appropriate theory to apply to malware architects. As such, this thesis explores a variety of complex issues, with the intention of gaining insights into why malware architects commit their crimes. This thesis is intended to help fill a gap in criminological knowledge regarding malware and provide a pathway to future research into malware and malware architects.

Declaration

This thesis and the research undertaken by myself has not been submitted for a higher degree to any other university or institution.

S.Morrison

Table of Contents

Chapter One - Introduction

In 1999, the world waited in anticipation as to the fate of the Internet, with fear that the Y2K millennium bug[1] would cause an ʼelectronic Pearl Harbor' (Smith, 1998). However, when the bug failed to deliver any form of destruction, our increasingly technologically reliant societies, and criminologists in particular, were left to come "to terms with the phenomenon of cyberspace [and the] potential harm of cybercrimes" (Wall, 2001 p. 2).

In 2019, 20 years later, the global economic impact of cybercrime is estimated to reach $2.1 trillion USD, which is four times more than the global impact of cybercrime in 2015 (Morgan, 2016). These figures may be conservative, however, with the World Economic Forum suggesting that a significant number of cybercrimes, particularly cyber-espionage, are still under-reported, indicating that the true cost of cybercrime may, in fact, be much higher (Morgan, 2016).

This paper will undertake case study research into three malware architects, the people who write malware. The primary goal of this research is to establish whether these 'architects' made a rational choice to commit crime.

Malware is a generic term, coined in a chat room by Yisrael Radai in 1990, to define a range of malicious code (Messmer, 2008). Today, the term malware is used globally to refer to crimes that involve a program designed to damage both stand-alone

[1] The Y2K bug was a foreseen problem with the calendar format on computer systems. It was feared that computer systems would not recognise the year '2000' and would revert calendar settings to '1900', thus causing massive errors and data loss. Patches were produced to fix this error and the problem was avoided (Encyclopedia Britannica, 2017).

computers or computers that are on a network (Swain, 2009, p. 1). This may occur via disruption, unauthorised access or financial damage, such as the cost to repair the technology malware infects and the financial loss occurred by the impact malware has upon business services. (The Parliament of the Commonwealth of Australia, 2010).

Malware consists of a variety of computer programs, each designed to by-pass information security protocols and/or cause damage. Malware referenced in this dissertation includes:

- Viruses – A virus is malware hidden within a program, such as Microsoft Word or Microsoft Excel, that spreads by replicating itself into other programs.
- Worms – Unlike a virus, a worm does not need a host program and will replicate itself across computer networks.
- Trojans – Trojans are malicious programs hidden inside a legitimate or perceivably legitimate program.
- Botnets – A botnet is a network of compromised computers that can be controlled simultaneously from a central point.
- Ransomware – This is a program that installs covertly on a victim's device, locking the owner out of the machine until a ransom is paid.

(DuPaul, 2012).

It is important to note that the actual writing of malware is not a criminal offence, unless the malware is "designed or adapted primarily for the purpose of committing any of the offences established in accordance with Articles 2 through 5" (European Treaty Series - No. 185, 2001, p. 5) of the international Convention of Cybercrime. This includes, the illegal access of a computer system, the illegal interception of non-public transmissions of computer data, the interference of data without right and the interference of systems without right.

The term architect will be used throughout this paper to describe writers of malware. A computer architect is someone who designs and configures an IT program or a system (Quora, 2016). As such, the term malware architect has been used when describing each of the case study subjects. The term architect is further relevant when examining case study research, where it will be demonstrated that the selected malware architects, in addition to designing malware, also designed ways to circumvent security protocols. This required a sophisticated understanding of the systems they were circumventing. For example, Robert Tappan Morris (RTM) had a deep understanding of Unix and previously identified security flaws in the Unix system. With this knowledge, he was able to design a program that was multiple-layered to by-pass security (Spafford, 1989).

From approximately 2003 onwards, organised crime groups began to use malware for financial gain, and as a result we can perceive that the malware architect's role also appears to change around this period (Bradbury, 2006). As demonstrated in Chapter Two, the introduction of multi-level support was introduced - the architect initially designs the malware, then various people in the organisation provide additional IT support, ensuring that certain elements of the malware work smoothly. Around this period, we also see that malware architects are not only designing computer programs, but also designing networks to support these programs.

As will be demonstrated in Chapter Two, malware appears to be a 'black hole' of inquiry for criminologists until the turn of the century. Because of this, computer science and information security literature has been utilised to bridge this gap. Literature sourced from the domains of computer science and information security, demonstrates the creation of the first malware by highly skilled programmers as well as by amateur computer enthusiasts, with the first malware written in 1970 in the form of a proto-virus[2]. The literature also demonstrates the scope of the threat that malware presented to computers and their users from the outset.

With the arrival of the new millennium, criminologists began to focus on the broad topic of cybercrime (Wall, 2012) and eventually different sub-categories of cybercrime such as malware (Holt & Bossler, 2016). An early example of this research is provided by Gordon and Ma (2003), who applied rational choice theory to hackers and malware architects. Using the theory of planned behaviour (TPB), Gordan and Ma (2003) examined the motivations of these offenders. The basis of their research was a survey that was issued to 'various sources'[3] (Gordon & Ma, 2003, p. 7) with their primary research population being the hacker community and virus writers[4]. The study demonstrated that virus writers were motivated by peer recognition and approval, with family and friends able to influence the virus writer's decision to offend (Gordon & Ma, 2003).

[2] A proto-virus is a proto-type of a virus. An example of this is Stein's (1970) Cookie Monster Code. In this example, the code had to be manually uploaded to each computer as it did not have the capability to spread by itself.

[3] There is no reference in Gordan and Ma's (2003) paper as to what these sources are.

[4] The results in the study specific to virus writers was primarily sourced from previous research conducted by Gordon (2000).

Gordon and Ma (2003) also found that hackers were self-motivated individuals and had a low tendency to offend based on the restraining influence of family and friends. The study concluded that further research was needed and that future studies should provide an objective criterion "for comparing and contrasting differences in motivations, beliefs, and attitudes, providing a baseline for designing appropriate interventions and reducing the occurrence of these undesirable behaviors" (Gordon & Ma, 2003, p. 11).

While an important early work in this area of research, Gordon and Ma's (2003) study has come under significant criticism. In particular, Holt and Bossler (2016) have described Gordon and Ma's (2003) research as overly speculative, and that due to this limitation new "research applying rational choice to cybercrime [is] sorely needed" (Holt & Bossler, 2016, p. 74). The decision to undertake research into rational choice theory and its application to malware architects is based on the gap identified above by Holt and Bossler (2016), as well as a general gap in criminological theory, as outlined in the literature review (see Chapter Two).

In undertaking this research, this dissertation will apply Cornish and Clarkes (1986) rational choice perspectives (RCP), as analysed in detail in Chapter Two, with the offenders' decision to create malware. According to Cornish and Clarke (1986) rational choice occurs when individuals weigh the potential risks against the rewards of participating in a crime in order to determine what their future actions will be. A rational choice perspective contrasts with the over-pathologising of criminal behaviour, as undertaken by traditional positivist criminologists such as Lombroso, who focused on the criminal ‘other’[5], thereby providing a more effective theoretical tool for analysing “criminal behaviour effectively and [devising] better crime-control strategies” (Cornish & Clarke, 1986, p. v).

There are three main components of Cornish and Clarkes (2001) RCP, “the image of a reasoning offender, a crime specific focus and the development of separate decision models for the involvement process and criminal event” (Newman, 1997, p. 7) Further, each word in the phrase ‘rational choice perspective’, holds significant meaning. ‘Rational’ refers to strategic thinking, processing and evaluating opportunities and alternatives. ‘Choice’ refers to the making of a decision. While ‘perspective’ refers to the reorganisation of theory and data designed to help understand criminal behaviour (Cornish and Clarke, 1986). As Newman et al. (1997) explain, there needs to be “better information on how criminals themselves perceive penal sanctions, environmental opportunities, rewards and costs” (Newman, et al., 1997, p. v) in order to create effective crime prevention and deterrence policies.

It is the hypothesis of the researcher that malware architects weigh up the potential risks and rewards before making a decision to firstly write malware and secondly, whether to release the malware. In other words, malware architects make a rational choice to commit crime. In addressing this hypothesis, this research will pose the following questions for each of the case subjects:

1. What was the malware architect’s intention in writing the malware?

[5] A person alien to the rest of society, driven by abnormal evil motivations (Cornish & Clarke, 1986).

2. Did the malware architect reach the goal they intended with the writing of their malware?
3. Did the malware architect put in place precautions to prevent themselves from being identified?
4. Did the malware architect rely on trans-judicial boundaries to protect them from law enforcement when the malware was released, (that is, were they living in a separate country to where the malware was released)?
5. What were the punishments/repercussions or likely punishment/repercussions of releasing the malware the architect wrote?
6. How was the malware architect caught, if in fact they were caught?
7. If the malware architect has not been caught, what is their current status?
8. What damage did the architect's malware inflict?
9. Was the damage caused by the malware intentional?
10. Did the malware architect show remorse for their crimes?

As stated, the primary aim of this research is to determine whether malware architects make a rational choice to offend. It is anticipated that if this research indicates that this is indeed the case, then further research into malware architects and rational choice theory will be undertaken as part of a larger PhD project.

The scope of this paper includes architects who write and then release malware, thereby committing a criminal offence. As such architects who write malware for research purposes, and do not release the malware, and malware written for the purpose of cyber-warfare (for example, Stuxnet[6]) will not be included in this study.

In undertaking research into rational choice theory and malware architects, this paper will be divided into seven main chapters, including this, the introductory chapter, and the conclusion. The following chapter, Chapter Two, is a literature review which will demonstrate, as suggested, that while malware is not a new threat, it has only been in recent years that criminology has begun to explore malware in any significant depth, first under the broad topic of cybercrime, and only more recently as a specific type of cyber offence (Wall & Williams, 2013). This chapter will also provide greater detail into Cornish and Clarkes (2001) RCP. Chapter Three will detail the research methods involved in the study, providing a rationale for the decision to undertake case study research and the selection of the case study subjects.

Chapters Four, Five and Six consist of case study analyses of three malware architects. The first, Chapter Four will examine RTM (Robert Tappan Morris, mentioned above) writer of the Morris worm, followed by Chapter Five's review of David Lee Smith (DLS), writer of the Melissa virus, and lastly Chapter Six will analyse Evgeniy Mikhailovich Bogachev (EMB), the alleged writer of the Trojans ZeuS and GameOver ZeuS (GOZ).

[6] Stuxnet was malware that targeted industrial control systems and is responsible for extensive damage in 2010 to Iran's nuclear program. It has been alleged that Stuxnet was designed by the United States and Israel to slow down suspected nuclear bomb development in Iran (Zetter, 2014).

The final chapter will discuss the key findings of this dissertation, and address the hypotheses and research questions.

Chapter Two – Literature Review

Malware has been an emerging threat since its beginning in the 1970s. An early example of this may be seen by Walker (1985), writer of the first Trojan horse in 1975. Walker (1985) described just how easy and efficiently his early malware spread across the organisation he worked for, and what the consequences would have been if the program he wrote had been malicious instead of a 'harmless game'[7] (Walker, 1985).

In 1983 computer scientist Frederick Cohen demonstrated the potential damage and impact of a computer virus under controlled circumstances at LeHigh University (Zetter, 2009). When the results of Cohen's test became known across the university, a ban was placed on all computer security experiments as the potential harm was seen to outweigh any benefits (Cohen, 1984). Five years later, a Massachusetts Institute of Technology (MIT) graduate student released malicious code, known as the Morris Worm, into the Internet. The purpose of the code, according to its creator, was to "gauge the size of the precursor Internet of the time - ARPANET" (Radware, 2017). The code worked by exploiting known vulnerabilities[8] in UNIX and by also guessing weak passwords (Radware, 2017). In December 1990, RTM, creator of the Morris Worm, was the first person charged under the US 1986 *Computer Fraud and Abuse Act* and was sentenced to community service[9]. By the 1980s it had become evident to the computer sciences and information technology sectors that critical network

[7] For more information on the Trojan see http://www.fourmilab.ch/documents/univac/animal.html.

[8] These vulnerabilities included sendmail, finger, and rsh (Radware, 2017). A finger command will allow you, at the most basic level, to locate user information such as login name. rsh is a utility that allows you to log onto a computer, whilst sendmail is the service used to send email on a unix system. For a detailed review of UNIX commands see Brown University's Computer Science Unix Commands Guide at http://cs.brown.edu/courses/bridge/1998/res/UnixGuide.html.

[9] *United Sates v. Morris* (1990). See Chapter Four for a detailed discussion on RTM.

infrastructure was vulnerable to cyberattacks, and that the potential for harm was also high given that digital technology had become increasingly embedded in everyday society (Bachmann, 2011).

However, it was not until the turn of century that criminologists appear to have started investigating cybercrime. Recent scholarly literature written by criminologists provides some insight as to why it took members of the discipline until this time to focus on non-terrestrial crime. According to Wall and Williams (2013), one of the reasons appears to be a lack of understanding on the part criminologists regarding the technology used to commit cybercrimes. The cyber realm traditionally ‘belongs’ to the computer sciences (Holt & Schell, 2011), whilst criminology traditionally sits within the social sciences (White & Haines, 2001). As a result, criminologists were not prepared for the technology-based crimes that emerged in the 1980s and 1990s. As Wall and Williams (2013) explain, the “emergence of the first generation of cybercrimes […] were met with inadequate criminological insight. Criminologists were simply not prepared for the digital re-engineering of terrestrial crime” (Wall & Williams, 2013, p. 262). As a result, criminological research often neglected cybercrime, as seen in the example of a major – and purportedly comprehensive – criminological text in Australia, *The Cambridge Handbook of Australian Criminology* (Graycar & Grabosky, 2002) which does not once mention cybercrime[10].

[10] Other examples of criminological texts that have a broad theoretical and/or crime focus, yet do not look at cybercrime, include: Fattah, 1997; Garland, 2001; White and Haines, 2001; and Smith, 2002).

Wall (2001) claims that very little was written about cybercrime by criminologists prior to 2000 due to uncertainties regarding the continued existence of the Internet. As outlined in Chapter One, there was much publicity and anticipation regarding the Y2K millennium bug. Specifically, the concern was whether there would be an 'electronic Pearl Harbor' that would suddenly end the world's reliance on technology (Wall, 2012).

Wall (2001) identified several additional obstacles that initially confronted criminologists when studying cybercrime:

1. A lack of statistical data;
2. Under-reporting from victims;
3. Varying offender profiles depending on the cybercrime;
4. The trans-jurisdictionality of cybercrime; and
5. Confusion, often fuelled by the media, regarding perceived risk versus actual risk of cybercrime.

(Wall, 2012, pp. 7-10).

As will be demonstrated,16 years later, many of the obstacles identified above by Wall (2001) are still relevant and continue to impede cybercrime research today.

The remainder of this chapter will examine the evolution of criminological research into malware.

2.1 First Evolution – Criminology begins to examine cybercrime, touching occasionally on malware.

As suggested earlier, criminologists began to focus on cybercrime around the turn of the century and, as shall be demonstrated, an interdisciplinary approach to cybercrime also emerged during this period. As Holt and Bossler (2016) were to write several years later, for criminologists to conduct useful research into cybercrime, they must first "avail themselves of research and data collection techniques from computer science and information security" (Holt & Bossler, 2016, p. 55).

In 1998, Grabosky and Smith provided a summary of money laundering and electronic funds transfer (EFT) as 'on-line crimes'. There was no focus however, on cybercrime as a distinct category of offences. (Grabosky & Smith, 1998, p. 49). In contrast, a year later, Taylor (1999) writing from a psychology perspective, provided a detailed account of the academic debate that encompassed the computer science community in the aftermath of the Morris worm (1988) and the first criminal proceedings against a malware architect, as will be discussed in Chapter Four. On one side of the argument, members of the computer science community believed that an unwritten ethical code amongst programmers[11] had been broken with the release of Morris' worm. As such the only option was legislated punishment. The other side of the argument contended that there should be space for malware writers to experiment and learn, and that educating programmers on the correct ethics of programming would prevent future incidents (Taylor, 1999).

[11] The ethical code implied by Taylor (1999) is that computer code should only be written to better enable technology and the use thereof, and not cause harm to other computers and/or persons.

Several years later, Taylor[12] (2001) contributed one of the first books that provided insight into the emergence of cybercrime from both a criminological and interdisciplinary perspective (Wall ed., 2001). Taylor (2001) wrote a comprehensive overview of cybercrime, including a review of the public perception of hackers, crackers and fraudsters. In addition, Taylor (2001) provided an exploration of three very specific malware attacks that drew media attention in 2000. These three examples were reviewed and used to demonstrate why cybercrime in general, and malware specifically, began to gain attention from scholars (Taylor, 2001). The first example reviewed was Worms Against Nuclear Killers (WANK). WANK was a form of hacktivism[13] designed to disrupt The National Aeronautics and Space Administration (NASA) space shuttle Galileo[14]. Although the malware was unsuccessful in stopping the launch of the shuttle, NASA claimed the worm cost up to half a million dollars in damages (Taylor, 2001, p. 66). The second was tales of a super-bug, a term used in the early 2000's to describe viruses and worms designed intentionally to do maximum damage to technology (Clarkson, 2002). In 2000 the *Observer* published an article entitled '*Coming to a screen near you*' which claimed that a super-bug was imminent and would do incalculable damage. The third was an impending cyberwar which had caught the media's attention due to growing fears of conflict between the USA and Kosovo (Taylor, 2001, p. 68). All three cases provided fuel for a discussion that would occur on malware.

[12] Taylor (1999 and 2001) is an example of the importance of interdisciplinary research, being a Communication Theorist who has provided several pieces on cybercrime related issues in numerous criminological publications (see for example Wall 2001).

[13] Hacktivism combines the act of hacking with political and/or social protest (Rouse, 2007).

[14] The protest was against the plutonium based power modules used in the space shuttle and concern that if the shuttle exploded this would lead to carnage for Florida (Denning, 2006).

It was also at this time that Wall (2001) identified the three generations of cybercrime. The first generation, known as low-end cybercrime, is described by Wall (2012) as cybercrime that “initially occurred within discrete computing systems and was characterised by the criminal exploitation of mainframe computers and their discrete operating systems” (Wall, 2012, p 95). Low-end cybercrimes were described as crimes that are aided by technology, such as bomb making. In this example, an offender may search online ‘how to make a bomb?’ and may locate this information. If the computer system was removed from the scenario the offender would have to rely on another means to find instructions for bomb making, so in this instance the computer simply aided the offender. The second generation of cybercrimes concern ‘opportunities for crimes across a global span of networks’ and involve crimes that have adapted with technology. According to Wall (2012) the second generation has also contributed to the “circulation of criminal ideas” (Wall, 2012, p 97). This includes the instructions for the manufacture and distribution of synthetic drugs and how to by-pass security protocols in mobile phones and television decoders (Wall, 2012).

As Wall (2012) explains, second generation cybercrimes are “effectively ‘traditional’ crimes for which entirely new globalised opportunities have arisen” (Wall, 2012, p 96). For example, an offender may send an email to a victim claiming to be a rich prince escaping a war-torn country in need of assistance. If the victim agrees to help the prince for a mere $10,000, then they would be rewarded with untold riches. The prince in this scenario is a fraudster and there is no ‘untold riches’. Fraud being a ‘traditional crime’ that has been changed significantly in terms of scope and impact since the advent of IT networks.

The third generation or high-end cybercrimes are referred to by Wall (2012) as ‘true cybercrimes wholly mediated by technology’ that appeared with the introduction of broadband at the beginning of the 21st century. As the third generation is “solely the product of opportunities created by the Internet, they can only be perpetrated within its cyberspace and are therefore *sui generis* (of their own kind)” (Wall, 2012, p 98). An example of a third-generation cybercrime given by Wall (2012) is spam email with virus attachments or the more recent examples of ‘blended threats’. An example of a blended threat is when a perpetrator uses malware to not only access a victim’s personal information, but the victim’s computer is also recruited using the same malware, as part of a botnet.

In addition to the above categorisation, Wall (2012) suggests that second and third generation of cybercrimes may be categorised into one of three behavioural groups of offences:

1. relating to the integrity of the computer system,
2. assisted by computers, and
3. which focuses upon the content of computers.

(Wall, 2012, p 98).

The purpose of the above categorisation is to ensure each offence is understood by criminologists and to ensure criminologists and the legal profession have the correct understanding of the offence.

2.2 Second Evolution – Criminology begins to differentiate between the varying forms of cybercrime, including malware. There is also a focus on offender profiling and offender motivation.

Between 2004 and 2008 there is a decline in the quantity of cybercrime literature written by criminologists. Wall and Williams (2013) argue that this is due to a downward trend in cybercrime related incidents and as such, a “slowdown in empirical and theoretical work from a criminological perspective.” (Wall & Williams, 2013, p. 409). Research that is available at this time, as will be demonstrated below, suggests a number of significant changes. Firstly, there appears to be more interdisciplinary research available; secondly cybercrime began to be broken into sub-crimes such as malware; and lastly, the literature also demonstrated a new focus on offender profiling and offender motivations.

An example of interdisciplinary research relevant to this time may be seen in Hinde’s (2004) paper which was innovative in providing insight into the growth of viruses and worms in 2004 and the increase of attacks against unprepared organisations. Hinde (2004) revealed that attacks targeting organisations occurred predominantly due to a lack of awareness of the risks posed by malware (Hinde, 2004). This issue was further addressed two years later in Broadhurst’s (2006) criminological focused article that explored the need for a global defence against cybercrime. Broadhurst (2006) argued that unless security measures were implemented and updated continuously they were basically useless (Broadhurst, 2006).

Broadhurst (2006) also raised awareness of the fact that gone were the days of the script kiddies[15] of the past, creating and distributing malware. Instead, malware architects were increasingly malicious in nature and/or financially motivated. A point reiterated several years later by Furnell (2012) who suggested that malware was written in the past by computer scientists with the intention of showcasing the malware[16] they wrote. Whilst the modern malware of today, such as Botnets, is designed to hide on computers, running in the background as quietly as possible, so the user is not aware of the existence of the malicious code. As Furnell (2012) wrote "the modus operandi [of malware architects] is no longer 'flash, bang and boo!' and more likely to rely on 'softly, softly and shush!'"

Wall (2007) provides an overview of cybercrimes and reasserted the categorisation of cybercrime into three main areas, as discussed above (Wall, 2007). Wall (2007) also provided some background as to who the original cyber-criminals were and how crime had changed with the introduction of computers. Furthermore, a breakdown of motivations for offenders was provided. Wall (2007) also touched on several individual strands of malware, commenting that the definition of a cyber offender is outdated and that the construction of a cyber-offender profile is problematic.

Writing seven years later, Broadhurst, Grabosky, Alazab and Chon (2014) provided a comprehensive examination of various cyber-offenders and their motivations via an examination of several cybercriminal case studies. This paper also analysed

[15] Script kiddie is a term used to describe individuals who use previously written or generated code that is not their own (SecPoint, 2017).

[16] For example, the Cookie Monster virus, the sole purpose of which was to prank fellow students.

cybercrime and how it sits within organised and ‘not-so-organised’ crime. As such, it provided some insight into who the e-criminal is. Even though this paper is not malware specific, it did demonstrate where malware architects would sit within an organised e-crime network (Broadhurst, et al., 2014).

One of the most recent scholarly books written on cybercrime is from Grabosky (2016) who analyses cybercrime in the past, present and future, providing relevant information on spam and virus development[17]. Grabosky (2016) also briefly examines phishing and malware and provided summaries of these crimes.

2.3 Third Evolution – Criminology begins to frame cybercrime under criminological frameworks.

With a surge of cybercrime from 2010 onwards, there is an increase in literature on cybercrime, including criminological research that attempts to categorise cybercrimes under new and existing criminological theories. This demonstrated a significant shift, as criminologists were no longer just scratching the surface when researching cybercrimes but were now trying to deeply understand the nature and offenders involved with non-terrestrial crimes. It is anticipated that this research paper will follow a similar path and provide a deeper understanding of malware and malware architects, whilst also attempting to understand malware architects using rational choice theory.

An example of a similar approach to cybercrime research was undertaken by Bossler and Holt (2011), who attempted to frame malware victimisation under Routine Activity

[17] This includes reference to the Morris worm, however predominantly references to malware begin from 2000 onwards.

Theory (RAT). Building on the earlier writings of Grabosky and Smith (2001), the study concluded that many forms of cybercrime victimisation occurred because of the lack of antivirus software or similar programs to protect the victim's computer (Bossler & Holt, 2011). Holt and Bossler (2013) further investigate the significance of RAT and cybercrime. The study questioned RAT as a relevant model, but concluded that the study did provide further evidence that cyber-deviance increased the likelihood of malware infection.

In contrast Holt and Bossler (2016) provide a comprehensive overview of criminological theories used to frame cybercrime. This includes a comprehensive study of RAT and its implication to malware victimisation, concluding that "it may be that phishing victimisation is driven by factors beyond individual risks" (Holt & Bossler, 2016, p. 71) and that further research was needed to determine the fit of RAT and phishing victimisation.

A further study that attempts to frame cybercrimes under new or existing criminological theory include Kigerl's (2009) study on deterrence theory, which demonstrated that the law had no actual impact on the occurrence of spam when examining a sample of email between 1998 and 2008.The results rendering the theory invalid with regards to malware, as deterrence theory implies an offender will be deterred from committing a crime by fear of punishment (Kigerl, 2012).

There have also been studies conducted into subculture theories of crime and social learning theory. As technology allows individuals to connect without the fear of social rejection and/or legal ramifications, the Internet offers individuals the platform to explore interests that they would not otherwise have explored (Holt & Bossler, 2016). With technology constantly evolving and changing and still having a massive impact on everyday life, a continued study of online subculture groups is imperative “to better understand how technology creates enclaves for individuals to engage in crime and deviance” (Holt & Bossler, 2016, p. 87). With regards to social learning theory, Holt and Bossler (2016) suggest that this was relevant at the start of the Internet due to the skill level required to commit cybercrime. Today however, the Internet has provided the answers and shortcuts for people to commit crime without a full understanding of technology. This feeds into Wall’s (2001) second generation of cybercrimes mentioned above, which suggests the internet has provided a way of ‘sharing criminal ideas’.

In 2015 Goldsmith and Brewer undertook research under a new theory they coined digital drift theory. This theory examines the way individuals are able to drift in and out of online criminal communities. In some respect, it ties in with subculture theories of crime, however it only covers the digital existence. Two conditions are necessary for drift theory to occur, affinity and affiliation, which the Internet provides according to the research (Goldsmith & Brewer, 2015).

In the same year, Van der Wagen and Pieter’s (2015) provided research into cybercrime and actor network theory. This study looked specifically at the use of Botnets and how it was their belief, that a hybrid approach to cybercrime was needed to fully understand cybercrime. This would comprise of not only investigating the humans involved in the crime but also the technology.

To date, research that specifically frames malware and malware architects under criminological theory does not appear to exist.

2.4 The Current Study - Rational Choice Theory.

Rational choice theory suggests that malware architects make a rational choice to write malware, and that this is not a spur of the moment decision due to the time it takes to plan and then write the complicated computer code that is malware. For example, RTM's worm consisted of 8 files and over 3,500 lines of typed computer code (Morris, 1990).

Contemporary rational choice theory was derived from classical theories of rationality which make up some of the "first sets of writings typically considered as criminology"[18] (Hayward & Morrison, 2013, p. 67). Rational choice theory was further developed in the 1980s[19] with Cornish and Clarke's (1986 and 2001) rational choice perspectives (RCP), with the intention of locating criminological findings "within a framework particularly suitable for thinking about policy-relevant research" (Cornish & Clarke, 1987b, p. 1). Cornish and Clarke (1986) did not believe that all criminal behaviour could be defined as rational, however, they believed that RCP could be applied to a large variety of criminal behaviour due to the detail and scope contained in the six propositions or hypotheses that formulated RCP:

1. Crimes are purposive and deliberate acts, committed with the intention of benefitting the offender.

[18] Classical criminology is seen as the first criminology (Hayward & Morrison, 2013).

[19] See for example, Clarke and Cornish (1985) economic analysis of criminal behaviour.

2. In seeking to benefit themselves, offenders do not always succeed in making the best decisions because of the risks and uncertainty involved.
3. Offender decision making varies considerably with the nature of the crime.
4. Decisions about becoming involved in particular kinds of crime (involvement decisions) are quite different from those relating to the commission of a specific criminal act (event decisions).
5. Involvement decisions can be divided into three stages – becoming involved for the first time (initiation), continued involvement (habituation), and ceasing to offend (desistance) – that must be separately studied because they are influenced by quite different sets of variables.
6. Event decisions include a sequence of choices made at each stage of the criminal act (e.g., preparation, target selection, commission of the act, escape, and aftermath).

(Cornish & Clarke, 2001, p. 24).

Cornish and Clarkes (2001) RCP's suggests that specific characteristics of each offence should be examined when determining the offender's decision-making process. This is due to the fact that each crime is different and therefore the approach an offender may take with regards to, for example, burglary will be different to that of malware. A burglar for example may assess the risks involved in robbing a house as not only a "potentially limited list of hard factual information, [but] also the hazards of the event" such as residents being home (Cornish & Clarke, 1987b, p. 44) In comparison, the choice-structuring properties of malware architecture and subsequent spread of the malware, as outlined in Chapter Three, suggest that overall, there is minimal risk of detection by information security experts and/or law enforcement. As will be demonstrated, very few malware architects have been identified since malware was legislated against in 1986 and, unlike a home burglar, malware architects work from the comfort of their own homes. Therefore, it may be hypothesised that the overall risk to malware architects of being identified and then apprehended is low.

Particular to Cornish and Clarke's (2001) RCP's are the notions of involvement process and the criminal event. The involvement process includes the background factors seen to influence a person or persons decision to offend. Involvement processes are often multistage and can extend over long periods of time. For example, the involvement stage for someone committing a burglary, according to Cornish and Clarke (1986) may include the offender's:

1. Background factors, such as their upbringing, social status and temperament.
2. Previous experiences with crime and law enforcement agencies.
3. Generalised needs, such as money.
4. Evaluation of solutions to the crime, such as the reward versus the punishment.
5. Perceptions regarding legitimate solutions versus illegitimate solutions.
6. Reaction to chance event, such as the ease of opportunity.
7. Readiness to commit the offence.

In contrast, the criminal event is a shorter process that involves the offender's decision to commit the offence and utilises "circumscribed information largely relating to immediate circumstances and situations" (Cornish & Clarke, 1987b, p. 2). With regards to burglary it may include a decision to offend in a wealthy neighbourhood, choosing an unoccupied house instead of a house were residents are sleeping, a house that is not visible from the street as opposed to a house that is, and a house that has no guard dog in contrast to one that does (Cornish and Clarke, 1987b).

Chapter Three - Research Methods

As demonstrated in Chapter Two, malware is a new area of study within criminology. In addition, as will be demonstrated, malware architects are 'hard-to-study groups' with very few malware architects ever having been identified (Drozhzhin, 2015). In order to test whether rational choice theory may be applied to the offender's decision to create and distribute malware, a qualitative research approach will be adopted. This is because qualitative research methods "have their greatest appeal when we need to explore new issues, investigate hard-to-study groups, or determine the meaning people give to their lives and actions" (Bachman & Schutt, 2015, p. 171).

There are a number of methodological complexities to consider when studying cybercrime, and variants of cybercrime, such as malware. An example is the general lack of data available on cybercrime (Wall, 2012, p. 7). This is the 'dark figure' of crime, all of those unrecorded and undetected offences which are not included in official crime statistics (Noaks & Wincup, 2004). Malware is often unreported and undetected, with variants of malware, such as botnets, typically designed to remain invisible to the victim[20] and, by extension, to law enforcement agencies (Drozhzhin, 2015).

As noted above, the identification of malware architects is not common. In 2004 the arrest of 18-year-old German student, Sven Jaschan architect of the Sasser worm, was reported as 'noteworthy' as it was the first time in three years that a malware architect had been arrested for a major malware outbreak (Sullivan, 2004). During this

[20] Not all malware has been designed to remain undetected, with earlier variants of malware intentionally drawing attention to itself, and in some instances becoming interactive with the infected user. For example, the Casino virus required infected users to play a game of blackjack to gain access to files the virus had hid (Dellinger, 2016).

time, a number of other significant worms had been written and released causing colossal infection rates, with none of the authors to date having been arrested or even identified. This included, Blaster Worm (2003) which infected over 100,000 Microsoft computers (Technopedia, 2017), SQL Slammer (2003) which infected an estimated 250,000 computers within 24 hours (Knight, 2003) and Code Red and Code Red II (2001-2002) which infected over 300,000 computers (Left, 2001).

Internet worms, like the ones mentioned above, appear to have been common in the early 2000's. However, by 2003 there appears to be a shift in who is writing the malware and the type of malware that is being written.

> *The reason is twofold, explains Sophos' Cluley. Firstly, sending out a rapidly proliferating worm to create a huge botnet is too obvious and raises too many alarms, prompting users to take security measures. Yesterday's hobbyist malware writer was generally an adolescent male wanting to be noticed by his peers. Today's for-profit malware writers want to stay under the radar, because if their product is noticed it prompts victims to take action and reduces the number of compromised machines. […] Organized commercial malware authors want to enslave, not destroy, their targets.*
>
> (Bradbury, 2006, p. 90).

Previously malware had been written for non-commercial reasons such as fame and boredom. However, by 2003, the majority of malware had become financially motivated, coming out of the Far East and Russia where cybercrime laws were/are more lenient or non-existent. This placed further burdens on law enforcement to identify malware architects and for security experts to identify malware (Bradbury, 2006). For example, a common practice which emerged at this time was the registration of thousands of domain names worldwide through numerous registrars by criminals utilising botnets. The botnets were used as proxies to illegal websites which were kept in ‘hibernation’ until activated for use. Hibernating proxies, which equate to approximately 75% of botnets, are nearly impossible to find (Bradbury, 2006). Cross-border issues also made, and still make, it difficult to catch and prosecute malware architects. As Bradley (2006) explains, even if an architect is identified, often the case has to be dropped due to the fact that there is not enough evidence in one country alone to support the case.

The above paragraphs have demonstrated the lack of statistical data available on malware, the elusive nature of malware architects, and the unavailability of known malware architects to survey. The critical necessities, according to Bachman and Schutt (2015) to the application of quantitative research methods. In fact, malware appears to be, one of those areas of criminological inquiry that is just “difficult to investigate using official data and survey methods” (Noaks & Wincup, 2004, p. 11). As such a qualitative research approach has been chosen for this study, qualitative being the most appropriate approach to criminal phenomenon that is new and relatively poorly understood (Bachman and Schutt, 2015).

Creswell and Poth (2018) categorises qualitative research into five broad categories as defined below:

- Ethnography – With roots in anthropology this involves the researcher immersing themselves within a particular culture or subculture. However, ethnography in criminology can raise a number of ethical concerns, such as witnessing criminal activity[21].
- Narrative - This tells the story of one or two individuals and relies heavily on in-depth interviews, with the intention of the results providing insight into how this person or persons life can provide an understanding to their greater community.
- Phenomenological - Such studies rely on a combination of research methods such as reading documents, conducting interviews and visiting events. The researcher is looking to the nature or essence of an event or activity.
- Grounded theory - In contrast to phenomenological studies, grounded theory attempts to answer why an event occurred, relying heavily on interviews and document analysis.
- Case study research – This relies on multiple data sources, such as court documents, newspaper reports, biographies, in order to understand a crime. It may or may not include interviews.

As seen above, the first four qualitative research methods, ethnography, narrative, phenomenological and grounded theory, would rely on interviews or interactions with malware architects. As already discussed however, interviewing known malware architects does not seem plausible at this stage due to the elusive nature of this cohort of offenders. Therefore, a qualitative research approach employing three case studies,

[21] See Yates (2004) for further details.

outlined in the introduction has been chosen for this study. The three case study subjects, RTM, DLS and EMB were chosen for the following reasons:

RTM

RTM is the architect of the 1988 Morris worm and the first person in the United States to be arrested and convicted for the creation and distribution of malware (Hafner & Markoff, 1995). The case of RTM and the Morris worm has been selected for this research as it is an early example of malware architecture and the first recorded case of a criminal conviction against a malware architect. As such, both RTM and his worm[22] have been written about extensively in both computer science and information security research. There was also extensive newspaper coverage of RTM's prosecution and appeal, as well as court documentation on RTM's case.

DLS

DLS is the author of the 1999 Melissa virus, one of the first viruses to spread via email and to rely on social engineering (Holguin, 2002). News reports would suggest that DLS did not have malicious intentions when writing the virus and simply did it 'for laughs' (Holguin, 2002). However, the very existence of Melissa placed extensive burden on email servers, in turn creating a global DOS type attack. Melissa infected over 100,000 computers, causing an estimated $1.1 billion in damage (Holguin, 2002). DLS was chosen as a case study as there is extensive information available on both him and the Melissa virus from the information security and computer science

[22] The identification of RTM'S program as a worm and not a virus is disputed (see for example Eichin & Rochlis, 1989a and b and Seeley, 1989). For the purpose of consistency with the title of the malware 'Morris worm', this paper will refer to RTM'S program as a worm.

community. In addition, information pertaining to DLS's defence and plea hearing as well as press releases from the U.S. Department of Justice are readily available.

EMB

The final case study will examine EMB, aka Slavik, aka Lucky12345. EMB is wanted by the FBI for the creation of a sophisticated botnet designed for banking fraud and ransomware (Krebs, 2015). According to the FBI and FoxIT, an information security firm that investigated EMB for a number of years, EMB wrote and managed as a service the malware ZeuS and GOZ and deployed a ransomware known as CryptoLocker (Sandee, 2015). It is unknown whether EMB wrote or simply utilised CryptoLocker to further his criminal enterprise (Sandee, 2015).

Research for this case study will be conducted primarily from secondary sources, with extensive information being available through investigations conducted by information security specialists and the FBI. There are also documented online discussions in chat-rooms by EMB giving detailed information on Zeus and GOZ.

EMB was chosen as a case study as ZeuS and GOZ are contemporary examples of highly sophisticated malware. In addition, there is ample information on the notorious programmer due to the damage both ZeuS and GOZ caused for almost a decade.

In addition to the above, a number of other factors contributed to selecting RTM, DLS and EMB, namely:

1. The three cases represent an extensive timeline of malware writing. The Morris worm was created in 1988, the Melissa virus in 1999 and the final examples, ZeuS and GOZ were written in approximately 2006 and 2010 respectively (Krebs, 2015).
2. Each malware caused a substantial amount of damage at the time of their release[23].
3. All three cases, as will be demonstrated in each of the chapters, represented a new stage in the development of malware. For example, Morris' worm demonstrated the destructive nature of malware, the Melissa virus demonstrated a new way to spread malware through social engineering[24] and email, whilst ZeuS and GOZ are contemporary examples of an organised criminal network employing malware for financial gain.
4. As the research sections will demonstrate, there is a diversity and richness of information that has been written regarding all three of the case subjects.

Having decided upon a qualitative research project in the form of three case studies, the next stage was to determine which specific approach to employ. Stark (2005) identified three types of case study research: intrinsic, instrumental and collective (also known as multiple). In contrast to instrumental, the primary focus of intrinsic case studies concerns the unique circumstances of the case in question, for example, what

[23] The damage each of the malware created is relevant to the time of the release of the malware. For example, the damage the Morris worm caused was substantial for 1988 with an estimated 6,000 computers being infected (Hafner & Markoff, 1995), in comparison to the damage in 1999 with the Melissa virus where up to 100,000 computers were said to have been infected (Cluley, 2009).

[24] As Rouse (2016) explains, "[s]ocial engineering is an attack vector that relies heavily on human interaction and often involves tricking people into breaking normal security procedures" (Rouse, 2016b, p. 1).

led Eric Harris and Dylan Klebold to kill 13 people and themselves at Columbine High School, in the United States of America (USA) in 1999 (Bachman and Schutt, 2015). In this study, the primary goal was to identify factors pertaining to Harris and Klebold's motivations and mindset, with an understanding that the findings and circumstances that led the two teenagers to kill would help researchers to better understand the circumstances of that particular case (Bachman and Schutt, 2015).

The primary goal of this research is to explore the complex issues surrounding malware architects, and to gain insight into the motivation and causation of malware writing. Once this information has been obtained, the results will be used to explore whether rational choice theory may be applied to the criminal behaviour associated with malware. This is therefore an instrumental approach to case study research, where a "generalisation of findings [is applied] to a bigger population" (Zainal, 2007, p. 4). As this research will explore more than one case study, this research will extend from an instrumental case study to a collective, instrumental case study.

Rational Choice Theory

Like all theoretical approaches to the study of crime, rational choice theory has both advocates and critics. Hayward (2011) believes, for example, that rational choice theory does not sufficiently recognise the irrational factors that affect offending, and that today's consumer culture feeds into the same irrational emotions seen in criminal behaviour, such as impulsivity, desire, immediate pleasure and also control (Hayward, 2004 and Fernell, 2006). This is examined further in Ferrell and Ilan (2013) who claim that, "for those who can afford to spend money gained 'legitimately' on the commodified thrills, pleasures and excitements offered by consumer culture, there may be less impetus to offend; for those who lack the means, criminality may offer the opportunity to attain these well-marketed emotional states" (Ferrell & Ilan, 2013, p. 373).

In contrast, Bachmann's (2008) application of rational choice theory to the hacker underground and in the broader sense, to cybercriminals, demonstrated that cybercrimes often involve premeditated decisions made by people he concluded to be rational offenders. Furthermore, Bachmann (2008) determined that the very nature of cybercrime requires careful consideration, research, design and execution. As Bachmann (2008) wrote,

> *Cyberspace exhibits several features that can lead a rationally-acting person to the conclusion that committing crimes in this environment is a viable option. Among them are the reduced risk of being apprehended, the multitude and instant accessibility of suitable, unprotected targets, and the high degree of anonymity.*
>
> (Bachmann, 2008, p. 47).

Bachmann (2008) concluded that cybercrimes are the result of calculated rational choices that take into consideration the offenders' self-interest, as well as the low-cost and low-risk opportunities the offences provide.

An underlying principle of rational choice theory, which is evident in the above study, is that crimes are chosen and committed for specific reasons and that a decision to offend is based on both the offence and the offender and the interactions between the two (Cornish and Clarke 1987a). Thus, “the final decision to become involved in a particular crime is the outcome of an appraisal process which, however cursory, involves evaluation of the relative merits of a range of potential courses of action. This process comprises all those thoughts likely, in the offender's view, to achieve his or her current objective” (Cornish & Clarke, 1987a, p. 935). Further, that it is imperative to understand the considerations of the offender and the properties of the offence, which structure the choices of the offender. As mentioned above, this term has been dubbed choice-structuring properties (Cornish & Clarke, 1987a) and has been applied when defining the research questions below.

As Table One demonstrates, upon determining the choice-structuring properties[25] of malware writing – or the offender’s ‘pros and cons’ list – research questions were formulated. The final step of this process was to construct hypotheses for the research. As Lammers’ (2004) study guide into developing research questions states, once a review of literature is undertaken and research questions established, the next stage of the research project is to “make one or more predictions for your study […] an educated guess regarding what should happen in a particular situation under certain conditions” (Lammers, 2004, pp. 4-5).

As this dissertation will follow the theoretical underpinnings of RCPs, each of Cornish and Clarke’s (2001) propositions outlined in Chapter Two were examined and a

[25] For examples of why malware is written see Gordon and Ma, 2003 and The Parliament of the Commonwealth of Australia, 2010. For information on anonymity see Goldsmith and Brewer, 2015. For examples of the disadavantages of writing malware see also The Parliament of the Commonwealth of Australia, 2010, Williams, 2012 and Brenner, 2012.

decision was made as to their relevance to this study. As each of Cornish and Clarke's (2001) propositions could be aligned with the hypotheses that were formulated for this dissertation, a decision was made to apply all six of the propositions. The results of this process are presented in Table One below.

Research Matrix:

Hypothesis	Clarke & Cornish's (2001) RCP (pg. 24)		Choice-Structuring Properties	Research Questions
H1. The malware was written for a specific purpose.	1. Crimes are purposive and deliberate acts, committed with the intention of benefitting the offender.	5. Involvement decisions can be divided into three stages – becoming involved for the first time (initiation), continued involvement (habituation), and ceasing to offend (desistance) – that must be separately studied because they are influenced by quite different sets of variables.	**Positives** *Financial Gain *To satisfy curiosity *Fame *Vendetta or obsession, such as stalking *Political stance/recognition *Protest *Anonymity of the internet *Trans-judicial **Negatives** *The offender may be identified *The offender risks punishment/repercussions if caught *Peer rejection *Malware writing and distributing is illegal and depending on the damage your malware creates could lead to jail time *If you are caught and currently living in a country without extradition you cannot leave that country for the rest of your life without the risk of being caught *You have no idea how big or how small your malware will be until it is released. This can go either way, the reward may not be worth the while or worse, the code goes wild and you lose control over it causing countless damage to infrastructure including hospitals.	RQ1. What was the malware architect's intention in writing the malware? RQ2. Did the malware architect reach the goal they intended with the writing of the malware?
H2. The malware architect failed to perceive the risks involved in releasing the malware.	2. In seeking to benefit themselves, offenders do not always succeed in making the best decisions because of the risks and uncertainty involved. 3. Offender decision making varies considerably with the nature of the crime.			RQ3. Did the malware architect put in place precautions to prevent themselves from being identified? RQ4 Did the malware architect rely on trans-judicial boundaries to protect them from law enforcement when the malware was released? RQ5. What was the punishment/repercussions or likely punishment/repercussions of releasing the malware they wrote? RQ6. How was the malware architect caught, if in fact they were caught? RQ7. If malware architect has not been caught, what is their current status?
H3. Malware architects did not know the extent of the damage the malware would cause.	4. Decisions about becoming involved in particular kinds of crime (involvement decisions) are quite different from those relating to the commission of a specific criminal act (event decisions). 6. Event decisions include a sequence of choices made at each stage of the criminal act (e.g., preparation, target selection, commission of the act, escape, and aftermath).			RQ8. What was the damage of the malware the architect wrote? RQ9. Was the damage caused by the malware intentional? RQ10. Did the malware architect show remorse for their actions?

Chapter Four – Robert Tappan Morris

The following paragraphs are intended to provide a cursory introduction into RTM. As previously explained, the involvement process constitutes any and all decisions taken by the offender to offend. As such, under Cornish and Clarke's (2001) RCP's, RTM's background and even the process he undertook in designing the worm is relevant in determining whether rational choice theory is applicable.

Introduction

According to Hafner and Markoff (1995), RTM was a talented programmer who claimed not to foresee the damage his worm would do once released. On the 2nd November 1988, RTM, a 23-year-old first year PhD student at Cornell University, released the Morris worm into what would soon be known as the Internet. The program, once uploaded, collected host, network and user information to break into other machines utilising known vulnerabilities in system software. RTM had coded a program that would scan for known vulnerabilities in Sun-3 and VAX computers running versions of Berkeley Unix (Seeley, 1989) and also by guessing weak passwords (Radware, 2017). The program would then exploit these vulnerabilities, replicate itself and repeat the process (Seeley, 1989).

The worm was a sophisticated program, with elements of the program still being used today by malware architects (Malenkovich, 2013). An investigation conducted by Cornell University would explain four months later, the worm "consisted of two parts: a 99-line "probe" written in high-level language […] and a much larger "corpus", which had been compiled into binary machine language" (Eisenberg, et al., 1989, p. 12), consisting of approximately 3,568 lines of source code in C (Eisenberg, et al., 1989). This suggests the malware took a long time and significant premeditation and planning to write. The worm would send out a probe which would attempt penetration of a computer, and if the penetration was successful, "would compile and execute itself on the penetrated host and then send for the corpus" (Eisenberg, et al., 1989, p. 12).

RTM testified that several hours after releasing the worm, he returned to his computer terminal at Cornell University to discover his worm had not worked the way he had intended. Computers were being affected multiple times, each time using more and more memory until the computer was rendered useless (Hafner & Markoff, 1995). Although the malware did not have a payload i.e. something inside the code that commands the worm do more than multiply itself, the malware did create a denial of service (DOS) type of attack,[26] a design flaw had caused the worm to spiral out of control. RTM testified that he was scared at this realisation and admitted to asking his friend to send an anonymous email explaining how to stop the spread (United States v. Morris, 1990). This also supports the proposition, discussed further down in this Chapter, that RTM wrote the program to test his skills rather than to cause malicious damage.

According to Hafner and Markoff (1995) however, the email did not work due to the havoc the malware was causing. Systems everywhere were either down or had been

[26] A Denial of Service (DoS) attack occurs when users are prevented from accessing computer systems, network resources or other devices due to malicious reasons (Rouse, 2016).

pulled off the Internet by system managers to prevent the malware from spreading. In the havoc the email was lost, not read or simply was not received until after the mayhem (Hafner & Markoff, 1995). It appears no other attempt was made by RTM to stop the malware once it was evident the program had not maintained the path he intended (Eisenberg, et al., 1989). During RTM's trial, he explained how he had asked a second friend for their opinion as to what he should do, but had chosen not to follow their suggestion as it was to write a program to counter the worm (United States v. Morris, 1990).

As a consequence of his actions, on the 25th July 1989, RTM was indicted on a felony charge of gaining unauthorised access to computers and causing more than $1,000 in damages (Hafner & Markoff, 1995). A little over a year later, on 22nd January 1990, RTM was the first person convicted for creating and distributing malware and sentenced to community service (United States v. Morris, 1990).

During RTM's trial, administrators from the University of California, the U.S. Army's Ballistic Centre, NASA, and the National Cancer Institute research laboratory testified that the Morris worm caused more than $150,000 in lost time and damages and infected an estimated 10% of all computers – equating to approximately 6,000 machines (United States v. Morris, 1990).

Researchers such as Spafford (1989) and Seeley (1989) have speculated that RTM gained knowledge to write his malware through research of his own and research from his father, Robert H. Morris (RHM). RHM had written about password security on Unix in 1978 and then conducted a joint paper on the general security of the Unix operating system in 1984. RHM was a formidable expert on Unix, having been one of the developers of the system (Eichin & Rochlis, 1989b). Then in 1985, as part of his work for AT&T Bell Laboratories, RTM wrote on Unix TCP/IP security, which covered the finger daemon and trusted hosts, two of the vulnerabilities the Morris worm exploited. RTM demonstrating that “the design of TCP/IP and the 4.2BSD implementation allow[ed] users on untrusted and possibly very distant hosts to masquerade as users on trusted hosts” (Morris, 1985, p. 1). This finding would also be applied to the Morris worm.

In addition to the above, it would appear that RTM invested a substantial amount of time researching and coding his program and made a rational choice to do so. During RTM’s trial, he admitted to spending approximately five days writing the initial program (United States v. Morris, 1990). Further time was invested in gathering encrypted passwords and researching security vulnerabilities. Part of the worm RTM wrote was a three-part password cracking program. *Crack_1()* was a list of common passwords discussed in the paper written by RHM, mentioned above. The second *Crack_2()* contained 432 words which appeared to be generated by RTM himself from password files he had stolen and cracked over time (Seeley, 1989). The third, *crack_3()*, attempted to crack the password using an online dictionary. According to Seeley (1989) this could have taken up to four months to have worked, if the worm had operated the way RTM had intended and went undetected. In addition, RTM made several attempts over a 13-day period to successfully execute the worm (Seeley, 1989). This suggests that RTM was continuing to modify his computer code in order for it to work. As such, it would appear RTM was continually making a decision to offend and that RTM's decision to finally release the code, or RTM's initiation (as per Cornish and Clarke's RCP's) into the criminal event was not done in the moment.

The remainder of this chapter will examine each of the hypotheses outlined in Chapter Three.

Discussion

H1. The malware was written for a specific purpose.

A review of the choice-structuring properties outlined in Chapter Three suggests that RTM's motivation or purpose for writing the Morris worm was to 'satisfy curiosity'. As demonstrated, RTM wrote a sophisticated computer program which comprised many elements. In answer to the first RQ1 - *What was the malware architect's intention in writing the malware*? - according to RTM, the worm was written to gauge the size of the Internet (United States v. Morris, 1990). RTM's intention was to have the malware sitting 'out in the ether' watching the growth of the Internet and reporting back to him (United States v. Morris, 1990). Evidence of this may be seen in the fact that, "the infection never spread beyond the Internet even though there were gateways to other types of networks" (Eichin & Rochlis, 1989a, p. 1). In addition, RTM's father in an interview suggested that his son was a talented programmer who was bored and over enthusiastic, and because of this RTM had programmed the worm to test his own skills (Markoff, 1990). This suggests that the purpose, of satisfying curiosity, contained two elements. The first to gauge the size of the Internet and report back. The second, it appears, was an exercise in testing RTM's own skills via exploiting known vulnerabilities, whilst working undetected using a complicated set of commands.

Based on the above, it would appear that the first hypothesis, that the malware was written for a specific purpose, has been answered in the positive. RTM invested substantial time in the planning and premeditation of committing the offence. Even if RTM mislead the court in his explanation as to why he wrote the Morris worm, which is possible, RTM made a rational choice to write a computer program that would circumvent security in order to watch the Internet grow. However, in answer to RQ2 - *Did the malware architect reach the goal they intended with the writing of their malware?* - it would appear that RTM did not. RTM had intended for his worm to go undetected. However, as demonstrated above, a flaw in his programming would have the opposite effect and would ultimately lead to RTM's arrest and prosecution.

H2. The malware architect failed to perceive the risks involved in releasing the malware.

In answering RQ3 – *Did the malware architect put in place precautions to prevent themselves from being identified?* - There is some evidence to suggests that RTM believed he would not be identified and that he did in fact put precautions in place to prevent his identity from being discovered. Spafford (1989) demonstrates that on discovery of the worm, computer scientists from Berkeley and MIT would take snapshots of the worm to analyse what the program did and how it did it (Eichin & Rochlis, 1989b). Initial analysis demonstrated whomever had written the program had also encrypted it. It is hypothesised that the code was encrypted to hide any markers in the code that may have led to the identification of the architect and to prevent anyone realising what the code was doing. However, the encryption used by the RTM was not very sophisticated, as Berkeley worked out how the malware operated and was able to stop it spreading within 12 hours of the malware being released (Spafford, 1989).

In addition, RTM testified that he intended his worm to ‘fly under the radar’ and not be detected. As discussed, RTM was charged under a section of law that needed to prove intention to use computers without authorisation (Pucci, 2016, p. 1). From the outset, the prosecution argued RTM knew he was doing something wrong as he released the program from MIT rather than Cornell, the university in which RTM was enrolled. Suggesting it was his intention to not only access computers without authorisation, but to conceal his identity as he knew he was doing something wrong (United States v. Morris, 1990).

In answer to RQ4 - *Did the malware writer rely on trans-judicial boundaries when the malware was released?* - The answer is no, RTM released the worm in the United States at a time when the Internet was primarily used by researchers and government agencies. As such, the malware appears not to have spread outside the United States (Marsan, 2008).

In answer to RQ5 - *What was the punishment/repercussions or likely punishment/repercussions of releasing the malware they wrote?* RTM was charged and found guilty under a section of law that “requires the Government to prove that he intentionally used computers without authorisation and altered, damaged or destroyed data in those computers, causing more than $1,000 in damage” (Pucci, 2016, p. 1).

The Court found that RTM intended to access computers without authorisation (United States v. Morris, 1990) and as such was found guilty and sentenced to 400 hours of community service and fined $10,000.00.

In answer to RQ6 - *How was the malware architect caught, if in fact they were caught?* and RQ7 - *If the malware architect has not been caught, what is their current status?* - A few days after the worm had been released, Markoff, a reporter from the New York Times, received an anonymous phone call stating that Mr X, writer of the malware, had not intended to cause damage and that is was a serious error of judgement (Hafner & Markoff, 1995). The caller was to call several times before referring to the programmer as RTM. Markoff conducted a search on the initials and came up with the name Robert Tappan Morris. This name was confirmed after Markoff spoke to RHM, chief scientist at the National Computer Security Center (NCSC), an arm of the the US National Security Agency (NSA) (Hafner & Markoff, 1995). RHM had not confirmed RTM's identity through intelligence gathered by the NSA, rather RTM testified that on the 3rd November 1988, he had contacted his father in his capacity as chief scientist at the NCSC to confess what he had done. Describing his experiment as a dismal failure, RTM explained how he had only intended the worm to copy itself once to each computer it come in contact with however, a design flaw had caused the catastrophe that followed (Hafner & Markoff, 1995).

There is some conflict in historical records regarding RTM's confession. During his trial, RTM would suggest that he had confessed to his father knowing RHM would use this information in his capacity as chief scientist at NCSC to undertake appropriate steps and inform the authorities (United States v. Morris, 1990). Newspaper reports at the time however, suggest that RHM forced his son to confess to the worm after RTM told his father in confidence that he was responsible for the mayhem (Malenkovich, 2013).

Based on the above information, it appears RTM failed to perceive the risks involved in releasing the worm.

H3. The malware writer did not know the extent of the damage the malware would cause.

With regards to RQ8 - *What was the damage of the malware the architect wrote?* - As demonstrated, RTM caused substantial damage by infecting 10% of all computers connected to the Internet resulting in over $150,000 in damages. In answer to RQ9 - *Was the damage caused by the malware intentional?* - When investigating whether RTM knew the extent of damage the malware would cause, the answer would appear to be no. As demonstrated above, RTM had a very good understanding of Unix security, but evidence suggests he was also learning as he went along. For example, RTM had made several attempts to upload the worm before he was successful. During the week of 19th October through to the 28th October 1988, Cornell mailer logs recorded attempts by an unknown person to access the Internet via sendmail[27], which is a popular email routing facility used by Unix (Rouse, 2006a). It appears that RTM

[27] This was not discovered however until a few days after the Morris worm hit, when Eugene Myers from the National Computer Security Centre analysed Cornell mailer logs (Seeley, 1989).

did not have a clear understanding of this technology[28] and was learning as he went along. However, RTM overcame the technical problems he was facing and on Wednesday 2nd November 1988 at 5:01:59 am, another test was conducted at MIT via a remote login from Cornell. This time the test was successful and the program, a form of malware, was executed on hosts connected to the Internet.

In examining the above sequence of events, inconsistencies occur with regards Cornish and Clarkes (1986) notion of criminal event which is defined as being limited to immediate circumstances. Evidence demonstrates that RTM made several attempts to launch the malware. With each failure, it appears RTM worked on his code to overcome the issues. As such, it appears the criminal event becomes complicated with malware offences, most likely as the distribution of malware was not a crime perceived at the time of writing the RCP's. Therefore, it is the position of this dissertation that more research is needed into the notion of criminal events with regards to malware.

Once the criminal event occurred, it appears the worm overworking the CPU was the key flaw with RTM's program, and the reason why it caused so much damage. This has been confirmed by computer scientists at both Berkeley and MIT whose investigation into the worm demonstrated that the worm had only removed files it had

[28] The attempts were unsuccessful (Eichin & Rochlis, 1989b). The most likely reason for this was the fact that a byte is made up of 8 bits, whilst simple mail transfer protocols (SMTP) only allow for 7 bit ASCII data transfers. As such, the most probable outcome would have meant that the SMTP server was removing the 8th bit, making the payload or the code that had been written unreadable by the service. Each byte can have a value of 0 through to 255. Removing the 8th, or most significant bit, means it can only be 0 through to 127 in value so the payload is unreadable.

created and had not deleted system files or any other files. (Seeley, 1989). The program was unable to identify when it had already infected a system, so the malware would repeatedly infect already infected machines, devouring processor cycles so that "on some computers there were hundreds of copies of the program running, slowing the machines to a halt" (Hafner & Markoff, 1995, p. 255).

During RTM's legal defence, he would argue that he wanted to ensure that the program did not copy itself more than once on a computer but also did not want anyone to prevent his program from executing. As a safety precaution, RTM programmed the malware to ask each computer whether it had received a copy of the program. To circumvent the issue of programmers outsmarting his code, RTM programmed a test where the worm would duplicate itself every seventh time it received a 'yes' response. Apparently, "Morris underestimated the number of times a computer would be asked the question, and his one-out-of-seven ratio resulted in far more copying than he had anticipated" (United States of America, Appellee, v. Robert Tappan Morris, Defendant-Appellant, 1991, p. 1).

In addition to the above, and as briefly explained, the idea of intent was examined at RTM's trial and in an investigation conducted by Cornell University. Both investigations concluded that it appeared RTM did not intend for the program to:

1. destroy any systems, files or data
2. interfere with normal functions
3. be discovered

4. spread out of control

The Cornell investigation likened RTM’s actions to,

> *driving of a golf-cart on a rainy day through most houses in a neighborhood. The driver may have navigated carefully and broken no china, but it should have been obvious to the driver that the mud on the tires would soil the carpets and that the owners would later have to clean up the mess.*
>
> (Eisenberg, et al., 1989, p. 7).

In response to RQ10 - *Did the malware writer show remorse for their actions?* - RTM during his trial was very apologetic for his actions (United States v. Morris, 1990).

Conclusions

The starting proposition of Cornish and Clarkes (1986) rational choice theory approach to explaining crime is that the offender seeks to benefit themselves through “their criminal behaviour; that this involves the making of decisions and of choices however rudimentary on occasion these processes might be; and that these processes exhibit a measure of rationality (Cornish & Clarke, 1986, p. 1). In answering the first hypothesis, it was demonstrated that RTM made a rational choice to write a program for the specific purpose of benefitting himself, hiding his identity for the purpose of protecting himself. In addition, RTM invested substantial time in planning and researching his worm.

A review of RTM's background suggests that the decision to write the worm came out of his interest in information security flaws he had previously researched. It may also be suggested, that RTM's passion stemmed initially from his father's involvement in the development of UNIX. As a consequence of this passion RTM made a decision to write the malware and was not coerced or forced to do so. The writing and subsequent release of the Morris worm was RTM's decision alone.

It would appear however, that RTM's decision making was constrained by his inability to see the possibility of his own failure to write fool-proof code. More precisely, RTM failed to perceive the risks associated with malware writing. People in general, have limited capacity to see the consequences of their actions (Cornish & Clarke, 2001). In this case study, the consequences of RTM's action were severe. It appears RTM miscalculated the effects of his worm, this may be because this was a new crime and as such the consequences of committing such a crime were still being discovered, the Morris worm being the first major incident of malware (Seltzer, 2013). As Cluley (2013) reminisces, "back then no-one could have predicted just how much of an impact malware and cybercrime was going to have" (Cluley, 2013, p. 1).

Secondly, it appears, RTM genuinely believed that his code would not cause harm and would 'fly under the radar'. This fits in with RCP3 "*Offender decision making varies considerable with the nature of the crime*" (Cornish & Clarke, 2001, p. 24). In this example, RTM's decision to commit the crime appears to be, that in releasing the worm, not only would people be unaware of its existence but it would harmlessly track the changing shape of the Internet, giving RTM, one would assume, a huge academic advantage.

In answering H3, an issue was identified with regards to Cornish and Clarke's (1986) definition of criminal event. Malware is a criminal offence and in many ways, is similar to other criminal offences in that it requires a victim and an offender. However, unlike a mugging, for example, if the malware architect is unsuccessful in their first attempt to execute the malware, they may try over and over again, with very little chance of being noticed, until after the event, as per the example of RTM. It is suggested that it is impractical for a mugger to grab at a person's bag, fail, take a moment to work out why they failed and then attempt to target the same location, or even the same victim, over and over again, without running a very high risk of being apprehended once the police where informed of their initial attempt. It is important to note, that in 1986 when the RCP's were being written, malware was still very new, so in all likelihood, malware was not considered in the equation of the RCP's. Therefore, further research is needed into malware's effect on the RCP 'criminal event'.

Lastly, this chapter has demonstrated a clear initiation into crime by RTM as per RCP5 - *Involvement decisions can be divided into three stages – becoming involved for the first time (initiation), continued involvement (habituation), and ceasing to offend (desistance)* – One would assume that if the Morris worm had not spiralled out of control, RTM would have continued his experiment or his habituation. However, the worm did spiral out of control and RTM's habituation and desistance occurred jointly with his surrender. A situation one would assumed is mimicked in other criminal occurrences, where a first-time offender is caught or identified after their first offence and therefore chooses not to offend again.

Chapter Five - David Lee Smith

As per the previous chapter, the following paragraphs will provide an analysis of this information regarding DLS and the Melissa virus in order to explore the involvement process.

Introduction

On the 26th March 1999 DLS was a computer programmer working as a subcontractor or ‘trouble shooter’ for AT&T labs, when he uploaded or ‘initiated’ Melissa to the alt.sex Usenet newsgroup from his apartment in Aberdeen, USA. Melissa, reportedly named after a stripper in Florida, was hidden in a Word document which promised a list of passwords to pornographic sites. Instead of the passwords, upon opening the word document, Melissa was enabled (Cluley, 2009).

Unlike the stereotypical malware writers perceived at the time, who were assumed to be young adolescents and teenagers, DLS, to the surprise of many, was 30 when he was arrested for Melissa (see Cluley, 2009 and Acohido & Swartz, 2008). It would soon be discovered that DLS was not new to writing malware and had adopted an online ‘bad-guy’ persona for the writing and spreading of viruses under the name Kwyjibo[29] (Acohido & Swartz, 2008). DLS had also created an online ‘good-guy’ persona, under the name Doug Winterspoon, who would help people clean up viruses created by Kwyjibo (Acohido & Swartz, 2008). Similar to hero syndrome or hero

[29] The name Kwyjibo comes from the Simpson Episode ‘Bart the Genius’ where the character Bart Simpson uses the word Kwyjibo to win a scrabble match. The moment of the win was captured in the Melissa virus as the words that would occur randomly in open word documents as described further in the chapter.

complex, it appears DLS created “a desperate situation which [he could] resolve and subsequently receive the accolades from” (Cross, 2014).

Once Melissa was executed, the virus would gain control over the standard Document_Open() macro. Melissa would then attempt to deactivate macro security. Macros allow you to embed functionality within documents so that you may automate repetitive tasks. For example, setting up a macro that automatically attaches your company’s letter head[30]. Melissa checked for the value ‘Level’ in the registry key: HKEY_CURRENT_USER\Software\Microsoft\Office\9.0\Word\Security. If Level was found, Melissa assumed it was running inside *Word 2000* and disabled the Security. If Level was not found, Melissa assumed it had infected *Word 97* and deactivated Word’s virus protection. Melissa used a flaw in Microsoft’s Visual Basic for Applications (VBA) that allowed users to program an executable function into documents. DLS used this flaw to program malicious functionality in the form of an email attachment (Garber, 1999). DLS had created a Word document with a macro and hid Melissa inside the macro.

In addition, upon opening the Word document, the virus would infect normal.dot[31] (Standler, 2002). Once the normal.dot file was infected, every Word document created based on the normal.dot template would also be infected (Panda Security, 2013). This meant, Melissa embedded itself into normal.dot so not only would new Word documents have a standard font, they also came with a virus. If the victim then emailed

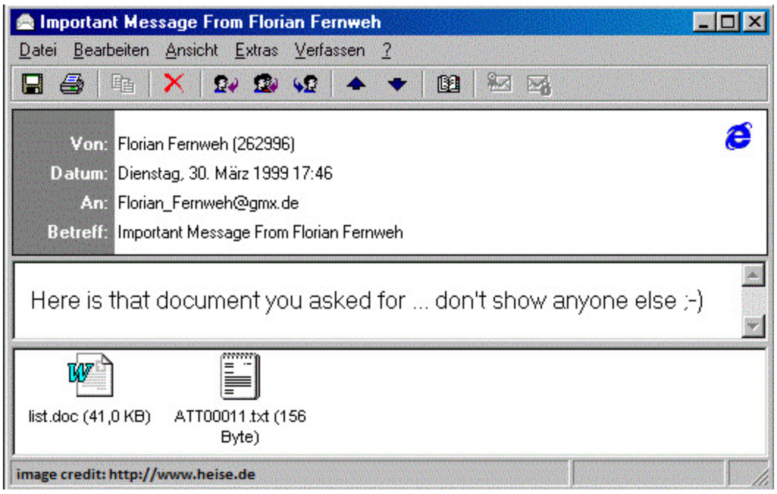
[30] For more information on Macro creation, see Sartain, 2015.

[31] Normal.dot is a Microsoft Word global template used to store default settings such as font and font size (OfficeArticles, 2014).

the document saved under the new corrupted version of normal.dot, the recipient upon opening the attachment would also become infected with Melissa.

As described above, Melissa, unlike the Morris' worm, did not spread automatically and relied on the user to spread the virus through such means as downloading an infected Word document, via infected external discs and drives, via Peer to Peer (P2P) file sharing and other online file sharing channels (Panda, 2017).

Melissa infected Microsoft Word 97, Word 2000 and NT and was designed to "infect macros in word processing documents" (Standler, 2002, p. 1). An innovative feature of Melissa was one of its Payloads. Melissa would spread itself via Microsoft Outlook by emailing itself to the first 50 addresses in the victims Outlook address book. Melissa would disguise itself as an important message from the victim (Standler, 2002), as seen below.


Important Message From Florian Fernweh
Datei Bearbeiten Ansicht Extras Verfassen ?

Von: Florian Fernweh (262996)
Datum: Dienstag, 30. März 1999 17:46
An: Florian_Fernweh@gmx.de
Betreff: Important Message From Florian Fernweh

Here is that document you asked for ... don't show anyone else ;-)

list.doc (41,0 KB)
ATT00011.txt (156 Byte)
image credit: http://www.heise.de

(Panda Security, 2013, p. 1).

As illustrated in the above screen capture, Melissa used a technique called social engineering, tricking people into thinking they had received some private or secret information, in order to tempt the user into opening the attachment. According to Cluley (2009), this was the reason Melissa “spread like wildfire around the world” (Cluley, 2009, p. 1).

Unlike the Morris worm, Melissa was not difficult to create. Although it appears DLS never admitted to how much time he spent writing Melissa, a review of the source code demonstrates 98 lines of code[32] as opposed to over 3,500 lines of code to write the Morris worm plus probe. This does not mean that Melissa could have been written as a spur of the moment decision. DLS still had to type the 98 lines of code, steal an email account and upload Melissa to the alt.sex user group.

Melissa worked by attacking a weakness in popular technology that was relied upon by so many (Garber, 1999). As Jeff Carpenter of the Pentagon-funded Computer Emergency Response Team, Carnegie Mellon University stated, Melissa was not the result of unusually brilliant computing. "What was interesting about this particular virus was not that it used groundbreaking [sic] technology," he said. “What made Melissa different from other viruses -- what made it breed so quickly -- was its use of e-mail, which has become one of America's favorite means of communication” (in Grunwald, 1999, p. 1).

[32] GitHub provides a full break-up of the source code with comments to how each section of the code works (darksheer, 2013).

The remainder of this chapter will examine each of the hypotheses.

Discussion

H1. The malware was written for a specific purpose.

After DLS's arrest, the press would suggest that Melissa was written just 'for kicks' (Cluley, 2009). DLS confirming this during his sentencing hearing when he stated Melissa was intended as a harmless joke (U.S. v. David Lee Smith, 2002). State Attorney General Peter Veriero argued however that DLS wrote Melissa primarily to see if he could (Grunwald, 1999). In reviewing the choice-structuring properties outlined in Chapter Three, taking into consideration all of the above statements, it appears DLS's motivation or purpose for writing Melissa, and the answer to RQ1, are akin with RTM's motivation 'to satisfy curiosity'. However, as described above, DLS had also adopted an online bad guy persona Kwyjibo, who took responsibility for writing particular malware, including Melissa. As explained in Footnote 29, DLS placed a reference to Kwyjibo in Melissa via a Simpsons quote, informing anyone who was paying attention that Kwyjibo was responsible for writing the virus. This would suggest a further reason for writing Melissa, the desire for 'fame', what Cluley describes as yesterday's hobbyist writing malware to be noticed by his peers[33]*.*

During sentencing DLS suggested that he intentionally wrote Melissa as a virus. However, he did not mean for Melissa to spread so fast and to cause so much damage. DLS would later argue he put in precautions to prevent Melissa from causing damage and that Melissa in retrospect had been a colossal mistake (U.S. v. David Lee Smith,

[33] See Chapter Three, page 29 for the full quote.

2002). Regardless of DLS's intentions, Melissa although "relatively benign compared with viruses that delete computer files or do other damage […] was the worst computer virus outbreak ever at the time" (Holguin, 2002, p. 1) as it clogged e-mail systems, causing some to crash (Holguin, 2002).

Rational choice theory assumes that offenders make a rational decision, weighing up the costs and benefits of committing crime (Newman, et al., 1997). It is the characteristics of the offence and nature of the offence that make it hard to believe DLS's did not contemplate that one of his viruses would spiral out of control. As Acohido and Swartz (2008) suggest, "some hackers would consider a couple of years in lock-up a small price to pay for securing a place in hacking lore" (Acohido & Swartz, 2008, p. 16). However, as DLS has not spoken openly regarding his motivations for writing malware, it can only be speculated that he was indeed seeking a place in 'hacking lore'.

Based on the above, H1 may been seen as answered in the positive. DLS made a rational choice to write a computer program that would exploit known vulnerabilities. This appears to be for his own satisfaction, to spread his own computer virus regardless of the consequences of that virus.

In answer to RQ2, it would appear that the malware spread as it was programmed to do. If fame was indeed a second choice-structuring property, DLS also achieved this. Melissa worked so well it caught the attention of the world. Within five days of the virus being released, DLS had been arrested. However, Cluley (2009), a respected

Information Security researcher and reporter has suggested DLS had actually “bitten off more than he could chew” (Cluley, 2009, p. 1), suggesting DLS simply made a colossal mistake.

H2. The malware architect failed to perceive the risks involved in releasing the malware.

In response to RQ3 it appears DLS put in precautions, albeit not very good precautions, to prevent himself from being identified. Investigations into Melissa, would reveal that the malware was uploaded originally from a stolen AOL account (Kocieniewski, 1999). In answer to RQ4, DLS uploaded the virus to the alt.sex usenet newsgroup from his apartment in Aberdeen, USA. However, Melissa was not restricted to the USA and spread globally, “overwhelming e-mail servers, before it could be stopped” (Garber, 1999, p. 16).

In response to RQ’s 5, 6, 7 and 8, on 1st April 1999, six days after Melissa was released, DLS was arrested at his brother’s house in Eatontown, USA under suspicion of committing ‘Fraud activity connected with computers’ (U.S. Department of Justice, 2002). As discussed, DLS used a stolen AOL account to upload Melissa to alt.sex newsgroup. However, it was this action that would ultimately lead to DLS’s arrest. It took the Attorney General's Computer Analysis and Technology Unit only three days of searching through thousands of AOL customer records to discover that the telephone line used to upload the original file was connected to an apartment in Aberdeen, Township, that would be confirmed as DLS’s apartment (Kocieniewski, 1999). This may suggest that DLS was not concerned with being caught or alternatively, DLS honestly believed Melissa would not spiral out of control. Regardless, as demonstrated, DLS did not put in adequate precautions to protect his identity.

DLS pleaded not-guilty to all charges. However, on the 9th December 1999 changed his plea to guilty (Standler, 2002). Facing 40 years in prison for writing and distributing malware, in an unprecedented turn of events DLS's sentence was reduced. DLS's defence attorney reported that this was due to DLS helping the police to "thwart and prosecute virus creators" (Holguin, 2002, p. 1).

It would later be revealed in court documents that DLS had provided the FBI with information that lead to the identification and arrest of OnTheFly, the Netherlands-based architect of the Anna Kournikova virus, Jan de Wit. Then in 2001 DLS claimed to have provided the FBI with further information that lead to the arrest of Simon Vallor (Cluley, 2009) architect of Gokar, Redesi and Admirer viruses (Leyden, 2003). An interesting point to highlight, even after his arrest DLS appears to have adopted a 'hero complex' by providing the FBI with information that only he had access to, that is, the identity of other malware architects.

Almost three years later, DLS was sentenced to 20 months in a federal prison and fined $5,100. Upon release DLS was prohibited to own or have access to an on-line computer, would undertake 100 hours of community service and have supervised release for the first three years. Although DLS was convicted under both State and Federals jurisdiction in the United States (Federal as the virus crossed state lines in the United States), DLS only served the Federal sentence. A press release from the U.S. Department of Justice announcing, “the state sentence is to run concurrently and co-terminously [sic] to the Federal sentence” (U.S. Department of Justice, 2002, p. 1).

H3. The malware writer did not know the extent of the damage the malware would cause.

As discussed above, Melissa caused extensive damage around the world, Microsoft suspending all incoming and outgoing email during the heart of the spread (Foley, 1999). In addition, it is estimated that Melissa caused over $1 billion in damages, spread world-wide and infected over 100,000 computers due to the astonishing speed that the virus spread (Cluley, 2009). Melissa also affected existing Word documents, via two means. The first, if the victim was infected with Melissa when “the number of minutes past the hour of the current time matched the date (for example, at 9:27 a.m. on 27 March), the virus would insert a Simpson’s quotation[34] (Grunwald, 1999, p. 17) in the active Word document the victim was working on. Second, Melissa would also send out random word documents to people in the victim address book, without the victim’s knowledge. This lead to sensitive documents being leaked (Grunwald, 1999).

In addition to the above, Grunwald (1999) suggested Melissa would become a beacon for criminal enterprise, highlighting the way malware would be carried out in the future. In 2016, *We Live Security* confirmed that Melissa influenced not only legitimate developments in viral marketing but also saw malware shift from “ego-gratification and

[34] “Twenty-two points, plus triple word score, plus 50 points for using all my letters. Game’s over. I’m outta here” (Grunwald, 1999, p. 17)

seeking of peer approval to full-blown, fully-monetized criminality” (Editor, 2016, p. 1). Melissa was responsible for not only financial damage, but damage to reputations, as well as highlighting the effectiveness of malware to criminal enterprises (Grunwald, 1999).

In terms of the criminal event, again there appears to be a distinction between the distribution of malware and traditional crimes, such as burglary. As Cornish and Clarke (2001) explain, “once motivated […] offenders become ready to commit a particular crime when they reach a decision that a valued goal will be more easily achieved using criminal rather than non-criminal means” (Cornish & Clarke, 2001, p. 58). Then once the crime occurs, the criminal event is complete.

In comparison, due to actions DLS took during the involvement process, the criminal event is not compressed into the one action of uploading a file to a news group. Once DLS uploaded Melissa, the virus spiralled out of control due to the social engineering tactics DLS employed. During DLS’s sentencing hearing, DLS would acknowledge that he intentionally designed Melissa so that “each new email greeted new users with an enticing message to open and, thus, spread the virus further” (U.S. Department of Justice, 2002, p. 1). A second comparison may be seen in the fact that DLS programmed Melissa to spread an additional 50 times with every computer it infected, using Microsoft Outlook. Lastly, Melissa would send out random Word documents to random recipients as well corrupt Word documents with script from a cartoon. In fact, even after Melissa was discovered and patches created to prevent its spread, Melissa would continue to be an issue as not every person patches their computers nor updates their computer to the latest Microsoft editions. Unlike a crime such as burglary, Melissa would keep on propagating long after DLS had been arrested.

With regard to RQ10, DLS intentionally designed Melissa to spread. That was the malware's purpose, as described under H1. Standler (2002) suggests DLS knew he was doing something wrong on the basis alone that he stole an AOL account and discarded the hard drives used to create Melissa, before hiding at his brother's house on the realisation that Melissa had spiralled. All of these events contradict DLS's attorney's initial description of Melissa being simply a form of cyber graffiti (McNamara, 2009). DLS apologised for his actions during his sentencing (U.S. v. David Lee Smith, 2002), however, no other evidence suggests DLS was remorseful.

Conclusion

As discussed, Cornish and Clarke (1986) define 'rational' as the strategic thinking of an individual who is processing and evaluating their opportunities and alternatives. Comparing this statement to DLS's action it would appear he made a well thought-out rational decision. Firstly, DLS designed Melissa and secondly, "using a stolen [AOL] account he posted an infected document on the Internet newsgroup 'Alt.Sex'" (U.S. Department of Justice, 2002, p. 1). The posting promised a list of passwords to pornographic websites, instead, upon downloading the word document Melissa executed.

Unlike RTM, there appears to be a number of motivations behind DLS's intention to write Melissa. As described above, it was suggested that DLS wrote Melissa for laughs, as a joke, to see if he could, and as a habitual offender. Rational choice theory suggests that background factors are seen to influence decisions and judgements to offend. For this case example, the decision to offend may simply have been an ego trip. Regardless of DLS's motivations, it seems unlikely that DLS could have failed to perceive the risks associated with releasing Melissa. Unlike the Morris worm in 1988 when malware was still new, Melissa was one of many, so the idea of Melissa spiralling out of control would have been a real possibility. As such, it may be assumed that DLS, a person of reasonable intelligence, could have assumed the possibility of Melissa not following the path he intended.

RCP states that not only is a crime specific approach required but also some crimes need to be broken down even further in order to "identify fruitful points of intervention" (Cornish & Clarke, 1986, p. 2). An example given by Cornish and Clarke (1986) is burglary and the need to distinguish between not only commercial and residential forms of burglary but also burglary in middle-class suburbs, public housing, and wealthy suburbs. As demonstrated, there are substantial difference between the Morris worm and the Melissa virus which may suggests finer distinctions need to be made with regards to malware. This point will be examined further once a review of the final case study is undertaken.

With regards to DLS's involvement decisions, the path DLS took is almost identical to RTM. DLS initiation into crime was of his own choice and once Melissa spiralled out of control DLS's habituation and desistance occurred jointly with his capture.

Another important similarity between Melissa and the Morris worm is the effect both had on future malware. Like the Morris worm, Melissa “appeared at the back end of the previous century, but it certainly foreshadowed much of what has been in evidence in the 21st century” (Editor, 2016, p. 1). History would demonstrate that many of Melissa’s traits, like RTM’s worm, would become common practice in the malware that would follow (Acohido & Swartz, 2008).

Chapter Six - Evgeniy Mikhailovich Bogachev

As per the previous chapters, the following paragraphs will provide an analysis of this information regarding EMB and his Trojan's ZeuS and GOZ.

Introduction

ZeuS is believed to have been released in approximately 2005/2006, by a then unknown person, who would later be identified by computer security researchers as EMB. Zeus was a malware kit which had two key functions, the first, a 'man in the browser[35]' attack, the second, as a botnet (Graff, 2017). The man in the browser attack was capable of "infecting entire JavaScript frameworks that were utilized to social engineer the victim for information, and then on the bank side, automatically inserting and authorizing transactions" (Sandee, 2015, p. 4). Other functions of ZeuS included keystroke logging[36], POST data logging[37] and the ability to install other affiliated malware, such as clickfraud[38] (Sandee, 2015).

[35] Similar to a 'man in the middle' attack, a 'man in the browser attack' "intercepts messages in a public key exchange. The attacker then retransmits them, substituting bogus public keys for the requested ones" (Rouse, 2006b; 1). The browser attack is harder to prevent as it includes the security mechanisms within the user's browser (Rouse, 2006b).

[36] A surveillance technique that registers each key stroke made on the victim's computer (Rouse, 2017b).

[37] Hypertext Transfer Protocol (HTTP) enables communications between clients and servers. POST is a method used by HTTP to submit data. It's most common use is for submitting web-forms and uploading files (Internet Engineering Task Force, 2014).

[38] Clickfraud occurs when botnets are used to click on adds that are pay-by click. Each time the ad is clicked the advertiser is paid. In most instances, the owner of the computer is unaware that their machine is being used by a botnet to generate money (Beal, 2017).

In 2007 ZeuS was identified as malware involved in large scale attacks against banks. This resulted in information security researchers taking interest in ZeuS and its creator, including research from Fox-IT, the organisation that would eventually identify EMB (Sandee, 2015). It would be discovered that EMB had created a very profitable criminal enterprise around ZeuS. Using a technique called ‘webinjects’ and a hybrid attack model which circumvented two-factor authentication, EMB introduced new techniques that took malware to a new level of capability and efficiency. Like RTM and DLS, EMB was responsible for creating ground-breaking technology that has aided the development of malware and is still used by cybercriminals today (Sandee, 2015). ZeuS would soon become the most popular malware on the market due not only to paying customers, but also because of ‘piracy’ – other cybercriminals were sharing the software without paying a commission to EMB (Sandee, 2015).

In 2009 a number of events occurred which were significant to ZeuS. Firstly, in May 2009 the FBI began to investigate a number of large banking frauds. What was unusual is that in each of the cases the victims log-in and passwords were being used from the victim’s own IP address. $450,000 was lost in May by First Data Bank, followed quickly by $100,000 from First National Bank of Omaha (Graff, 2017). An examination of the victim’s computers would reveal they were both infected with the same malware, ZeuS.

In 2009 the business organisation of ZeuS also appeared to change. As Graff (2017) explains, EMB

> *"started cultivating an inner circle of online criminals, providing a select group with a variant of his malware, called Jabber Zeus. It came equipped with a Jabber instant-message plug-in, allowing the group to communicate and coordinate attacks... Rather than rely on broad infection campaigns, they began to specifically target corporate accountants and people with access to financial systems".*
>
> (Graff, 2017, p. 1).

In September 2009, the FBI discovered a server in New York which seemed to be connected to the ZeuS network. Upon examination of the server, it was discovered that it was a Jabber Server containing thousands of lines of chatroom conversations outlining ZeuS' whole operation, this included the use of money mules. The criminal organisation would hire new immigrants or students in various countries to open bank accounts. A few days later the mules would return to the bank and withdraw a large sum of money that had been deposited into the account via a fraud. The mules would keep a small percentage of funds for their part and give the rest of the money to their handler, the person who had made initial contact with them and provided them with their instructions (Graff, 2017).

Investigations by the FBI revealed this modus operandi in several countries including the United States, Czech Republic, the United Kingdom, Ukraine and Russia. As a result of the investigation, 39 people connected to ZeuS were arrested globally, including Vyacheslav Igorevich Penchukov, aka tank, EMB's second in command, who was residing in the Ukraine. The Jabber server also provided the FBI with a name

Slavik aka Lucky12345, as architect of ZeuS. Slavik, according to the Jabber transcripts was male, residing in Russia and married (Graff, 2017).

By 2010, it is believed that Slavik had begun to push his business towards organised crime, when he released an elite version of ZeuS which had been created for premium clientele connected to Russian organised crime syndicates (Graff, 2017). Unlike previous versions of ZeuS, this version had an encryption key that would be sold to clients for upwards of US$10,000 (Sandee, 2015). This ensured that only those clients who paid for the encryption key could unlock the program that had more functionality than previous versions of ZeuS (Sandee, 2015). By the end of 2010 however, the malware landscape appears to change with the introduction of a new rival malware called SpyEye (Sandee, 2015). In an unprecedented turn of events, Slavik announced that he had handed the source code of ZeuS to SpyEye's author Harderman, and as such, Harderman would now be looking after ZeuS's clientele (Sandee, 2015, p. 4). EMB had announced his retirement.

However, EMB's retirement appears to have been a ruse, as not long after, EMB re-emerged having written a new malware, GOZ. Unlike ZeuS, the management of GOZ had been set up as a sophisticated organised crime operation named the Business Club (Graff, 2017). The Business Club consisted of approximately 50 individuals, which included "24/7 tech support technicians, third-party suppliers of ancillary malicious software, as well as those engaged in recruiting money mules" (Krebs, 2015, p. 1). Unlike ZeuS, GOZ was designed as a tool that would only be used by the Business Club and not rented out or modified for use by other criminal persons.

GOZ worked by stealing banking credentials, draining the bank account clean and transferring the money overseas. Once the theft was complete, the Business Club would use GOZ's botnet to create a Distributed Denial of Service[39] attack (DDoS) against the victim's bank. This was for two purposes, the first, to distract the bank from the theft. The second, to prevent the victim from logging into their account to report the theft. Once the clearance time passed and the funds were released, the DDoS attack would end. GOZ focused on large dollar amount frauds targeting accounts that could accommodate six and seven figure transactions. An example of this may be seen on the 6th November 2012 when the Business Club successfully stole US$6.9 million from an undisclosed individual in a single transaction before hitting the bank with multiple DDoS attacks in order to distract the bank and the account holder from noticing the transaction (Graff, 2017).

GOZ was also designed to allow add-ons, such as ransomware. This was demonstrated in 2013 when the Business Club expanded its services and deployed a ransomware known as Cryptolocker, utilising the GOZ botnet to spread the malware. *Wired* (2017) describes Cryptolocker as the first mainstream ransomware with an estimated 250,000 machines worldwide being infected within the first year. Once again EMB had designed ground breaking technology utilising a service called Bitcoin[40] to

[39] A Distributed Denial of Service attack (DDoS) expands on the DoS attack by leveraging multiple systems to attack the victim making it difficult to defend against. To achieve this the attacker gains control over one system which is dubbed the master system and then uses this system to control other systems known as 'zombies' to form a botnet. "Once the botnet is assembled, the attacker can use the traffic generated by the compromised devices to flood the target domain and knock it offline" (Rouse, 2017a, p. 1).

[40] Bitcoin is an online payment method see https://www.bitcoin.com for more information.

anonymously cash in on his new business, extortion. EMB “didn’t charge an exorbitant amount, but he made a lot of money and created a new type of online crime” (Stone-Gross quoted in Graff, 2017; 4).

Research conducted in 2015 by Fox-IT would reveal that in 2013 EMB appears to have retooled one of his GOZ botnets for intelligence gathering purposes in the Ukraine. Sandee (2015) suggests this activity was kept separate from the Business Club and that only EMB knew about this side of the business. The botnet would look for “keywords in emails and documents that would likely only be found in classified documents” (Sandee, 2015, p. 3). It would also be discovered that the same commands would be used against intelligence services in Georgia and Turkey (Sandee, 2015). It was also during 2013 that investigations conducted by Fox-IT would reveal Slavik’s true identity as EMB (Sandee, 2015).

In late May 2014, a global operation dubbed ‘Operation Tovar’ took place. Between 30 May to the 2nd June 2014, law enforcement and security researchers successfully gained control of the GOZ botnet. Described as “cyber-hand-to-hand combat” (Graff, 2017, p. 4) officials battled with EMB to seize control. Tovar was carried out by agents from the FBI, Europol, UK’s National Crime Agency, researchers from VU University Amsterdam and Saarland University in Germany, as well as information security firms Dell SecureWorks, CrowdStrike, McAfee, Symantec and Trend Micro (Krebs, 2014).

Discussion

H1. The malware was written for a specific purpose.

In response to RQ1, it appears that the choice-structuring properties of EMB was specifically, 'financial gain through criminal means'. The FBI's most wanted page describes EMB as allegedly involved in "wide-ranging racketeering enterprise and schemes that installed, without authorization, malicious software […] on victim's computers" (Federal Bureau of Investigations, 2017, p. 1). The malware was designed specifically to capture bank account numbers, passwords, personal identification numbers and information pertaining to online banking acts (Federal Bureau of Investigations, 2017). A second choice-structuring property, as demonstrated above was for intelligence gathering purposes, as demonstrated through the redesign of one of GOZ's botnets.

In response to RQ2, ZeuS and GOZ have been reported as "the most successful botnet attack tools used by cybercriminals [with GOZ believed] to have enslaved between 500,000 to a million computers at its peak" (Stevenson, 2016, p. 1). GOZ consisted of 27 botnets, many of which were migrated from ZeuS, with approximately 200,000 infections active at any given time. The FBI has estimated that GOZ is responsible for more than one million computer infections and financial losses of more than $100 million (Federal Bureau of Investigations, 2017). As such, it is the conclusion of this dissertation that EMB achieved what he set out to do when writing his malware.

H2. The malware architect failed to perceive the risks involved in releasing the malware.

There is significant evidence to suggest that EMB was aware of the risks associated with malware intrusion. In fact, EMB put in place mechanisms to prevent his malware from being taken-off line. For example, EMB appears to have designed GOZ to be 'take-down proof' as it relied on both traditional command servers (which included a single command centre operated by an individual on a server) as well as peer-to-peer communications. If a command server was knocked off-line "the botnet owner could just set up a new server somewhere else and redirect the peer-to-peer network to it" (Graff, 2017, p. 3).

In answer to RQ3, and consistent with the previous two case studies, there are a number of factors that suggest EMB put precautions in place to prevent himself from being identified. Firstly, it took the FBI several years to notice ZeuS and then several more years to identify the author as 'Slavik' (Graff, 2017). Once Slavik was identified as the author, he staged his retirement in what appears to be a ruse intended to lead suspicion away from his next endeavour, GOZ.

The fact that it took officials outside of Russia[41] eight years to identify EMB as the architect of both ZeuS and GOZ, even after a global raid against ZeuS, also substantiates the view that EMB put precautions in place to protect his identity. In addition, it appears that versions of ZeuS, upon being released as open-source,

[41] It has been suggested that the Russian Government identified EMB and hired him as an intelligence asset in 2011. See Graff, 2017 for further information.

flooded computer systems around the globe (Graff, 2017). It is the hypothesis of this paper that this occurred so that when EMB released GOZ there would be less chance that EMB would be identified as the architect, with the expectation that GOZ would be mistaken as one of the many variants of ZeuS that were flooding computers worldwide.

In addition to the above, and consistent with most of EMB's behaviour, Fox-Brewster (2017) has suggested that cyber offenders residing in Russia remain free from government prosecution providing they follow three simple rules:

1. Do not commit crimes against Russia.
2. Share the wealth with Russia's Federal Security Service (FSB).
3. If you are asked to do a favour for the Russian government, do it.

Although there is no known evidence to suggest EMB paid money to the FSB, there is evidence to support that he followed rules 1 and 3. It appears the spread of ZeuS and GOZ was controlled through configuration files which told the malware where to spread based on IP-ranges (Stone-Gross, 2012), therefore ensuring the malware did not target Russia. In addition, EMB appears to have been providing intelligence to the Russian government[42]. After GOZ was taken down, Fox-IT (in Graff, 2017) provided a complete analysis of files captured demonstrating that:

- from 2011, GOZ tracked various geopolitical developments that affected Russia;

[42] See for example Schwirtz and Goldstein, 2017.

- in 2013 and 2014, keyword searches were conducted on English-language documents for terms such as ‘top secret’ and ‘Department of Defense’;
- further in 2013 infected computers in Turkey were instructed to conduct keyword searches on ‘weapon delivery’, ‘arms delivery’, ‘Russian mercenary’ and ‘Caucasian mercenary’; and
- in 2014, searches were conducted for documents from government officials such as Georgia’s foreign intelligence services and the Turkish Foreign Ministry.

In answer to RQ4 and RQ5 it appears that a number of persons connected to ZeuS and GOZ have been arrested for their role in the criminal network. However, all of these individuals resided outside of Russia at the time of their arrest (Sandee, 2015). Individuals identified as part of the Business Club or EMB’s other criminal enterprise who have remained inside Russia, including EMB himself, appear to have been actively protected by the Russian state (Krebs, 2015). Russia and the USA do not have an extradition treaty and the Putin administration has explicitly commented on EMB, stating that “they won’t arrest him unless they catch him committing crimes in Russia” (Estes, 2017, p. 1)[43]. However, in answer to RQ6, if EMB was arrested by the US, he would face up to 40 years in prison under the US’s 1986 *Fraud and Abuse Act*.

[43] There have been cases where Russian Police have arrested and charged cybercriminals. As suggested, in order for this to occur, however, the cybercriminals need to have attacked Russian citizens or businesses. For example, Russian Officers of the Interior Ministry and the FSB arrested members of a phishing gang based in St Petersburg after they allegedly stole $12 million ruble's from Russian Bank accounts (Zorabedian, 2015).

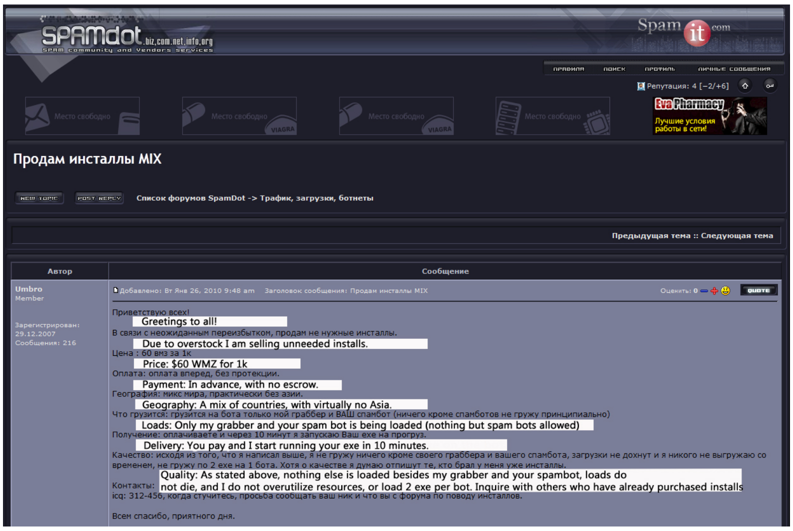
In answer to RQ7, EMB is still at large, however GOZ has been taken down. No evidence suggests at this time that EMB has created further malware since this take-down (Graff, 2017). In March 2017, newspaper reports indicated that EMB, now 33-years of age, had retired to southern Russia (Estes, 2017). As per Cornish and Clarke's (2001) RCP, EMB appears to have desisted his criminal career under his own terms.

H3. The malware writer did not know the extent of the damage the malware would cause.

The damage EMB caused differs greatly from that caused by RTM and DLS. Whilst the two previous case studies demonstrated malware that impaired the functionality of computers, ZeuS and GOZ were designed to go unnoticed for maximum financial impact. Both the Morris worm and Melissa virus were detected upon release, whilst ZeuS and GOZ were designed to operate in the background undetected.

In answer to RQ8 and RQ9, the malware EMB wrote was sophisticated, having caused substantial financial loss around the world. EMB relied on the use of botnets, which is the primary mechanism needed for the commission of most cybercrimes, such as, spam, DDoS attacks, child pornography distribution, click fraud, keylogging technology, mass identity theft and traffic sniffing (House of Representatives Standing Committee on Communications, 2010). Today "almost every major online crime may be traced to botnets" (Cyberspace Law and Policy Centre, 2009, p. 3).

In addition, ZeuS was successfully marketed for criminal purposes before EMB designed a more sophisticated version, GOZ. This demonstrates that unlike the two previous case studies, there is clear indication that EMB intended to write and release malware for financial gain. Research further indicates that based on the motivation of money and perhaps power, EMB created a criminal organisation around ZeuS and then GOZ. This idea of 'business' is reiterated in chatroom conversations where EMB is captured offering 'overstocked' installs to perspective clients (Krebs, 2012, p. 1).



Продам инсталлы MIX

Список форумов SpamDot -> Трафик, загрузки, ботнеты

Umbro
Member

Зарегистрирован: 29.12.2007
Сообщения: 216

Добавлено: Вт Янв 26, 2010 9:48 am Заголовок сообщения: Продам инсталлы MIX

Приветствую всех!
Greetings to all!
В связи с неожиданным переизбытком, продам не нужные инсталлы.
Due to overstock I am selling unneeded installs.
Цена : 60 вмз за 1к
Price: $60 WMZ for 1k
Оплата: оплата вперед, без протекции.
Payment: In advance, with no escrow.
География: микс мира, практически без азии.
Geography: A mix of countries, with virtually no Asia.
Что грузится: грузится на бота только мой граббер и ВАШ спамбот (ничего кроме спамботов не гружу принципиально)
Loads: Only my grabber and your spam bot is being loaded (nothing but spam bots allowed)
Получение: оплачиваете и через 10 минут я запускаю Ваш exe на прогруз.
Delivery: You pay and I start running your exe in 10 minutes.
Качество: исходя из того, что я написал выше, я не гружу ничего кроме своего граббера и вашего спамбота, загрузки не дохнут и я никого не выгружаю со временем, не гружу по 2 exe на 1 бота. Хотя о качестве я думаю отпишут те, кто брал у меня уже инсталлы.
Quality: As stated above, nothing else is loaded besides my grabber and your spambot, loads do not die, and I do not overutilize resources, or load 2 exe per bot. Inquire with others who have already purchased installs
Контакты:
icq: 312-456, когда стучитесь, просьба сообщать ваш ник и что вы с форума по поводу инсталлов.

Всем спасибо, приятного дня.

(Krebs, 2012).

With regards to RQ10, there is no indication that EMB showed any remorse for writing ZeuS or GOZ. In fact, the opposite appears to be true. For example, in January 2013, two security researchers were able to gain control of 99% of the GOZ botnet. However, within a week EMB had written a software update that saw control of the botnet back in his hands. In response, EMB contacted the researchers under his on-line name Slavik, mocking them for their efforts (Graff, 2017).

Conclusions

Research demonstrates that EMB displayed each of the attributes of a rational thinker. This is particularly evident when examining EMB's habituation or continued involvement in ZeuS and GOZ. EMB built an organisation around his malware, thereby demonstrating strategic thinking and processing. It also appears EMB worked either directly or indirectly as a Russian intelligence operative, utilising his botnet to spy on foreign governments to ensure the protection of the Russian State. This demonstrated the ability to evaluate and manage both opportunities and risks. Research further indicates that EMB did so of his own choice. For example, EMB chose to create a new improved malware, GOZ, when it appeared ZeuS had reached its end of life. The new improved malware was designed to be 'take-down proof' and generate more income than its predecessor.

With regards to legislation and policy designed to prevent the distribution of malware it appears to have not adversely affected EMB. This may be seen due to the trans-judicial nature of the crime, having released malware that did not target Russians and as such, avoiding breaking any laws in his own country. The only negative for EMB appears to be he can no longer leave Russia without risk of being arrested. As long as he resides in Russia, as he does at this point in time, he appears to be protected.

Clear limitations however, may be identified with Cornish and Clarke's (2001) RCP's with regards to EMB. Unlike RTM and DLS where information was readily available regarding their background and the apparent reasons they chose to commit malware intrusion, there is relatively little information available on EMB leading up to the release of ZeuS. Once ZeuS was discovered, EMB's involvement process could be mapped through his pseudonym's Slavik and Lucky12345 and then later under his real identity. However, a number of questions have been left unanswered, including EMB's initial decision process to write ZeuS as well as the hours it took to write both ZeuS and GOZ. The information that is available suggests that GOZ was run like a business with staff working 9 am to 5 pm to support the malware (Sandee, 2015). Investigations conducted by several security researchers also revealed that EMB made numerous updates and changes to both ZeuS and GOZ, suggesting that many hours went into writing and maintaining both sets of malware (see for example Krebs, 2015 and Graff, 2017).

A second limitation identified is in regard to the criminal event. Specifically, it is unknown exactly when ZeuS was released; Trojan malware is designed to go un-noticed in order to create back-doors into a computer system, to spy on the user to gain access to secure accounts such as banking, and/or to turn the victim's computer into a 'zombie' machine as part of a botnet (Lemonnier, 2015). With regards to ZeuS it appears EMB designed the program very well, meaning that it took computer security experts up to two years to discover ZeuS's existence. Upon discovery of the malware, computer scientists were able to put together a rough timetable that suggested the malware had been in operation approximately one to two years, however, an exact release date for ZeuS and GOZ is unknown (Graff, 2017).

In addition, with ZeuS being released as open-source code, any person could use the code or modify the code to suit their own purposes. As such, both EMB and ZeuS are responsible for facilitating multiple criminal events by multiple people. Without EMB providing a detailed statement as to when he released ZeuS and GOZ, and as to how many attempts it took him to release the malware successfully, little information is available on the exact details of the initial criminal event, what Cornish and Clarke (1986) refer to as the initiation. However, as per Cornish and Clarke's (1986) example of burglary, it appears EMB made an event decision when he calculated where to deploy the ZeuS malware, similar to a burglar choosing a house they perceived as a low risk job.

When comparing ZeuS and GOZ to the Morris worm and Melissa virus, a clear evolution of malware may be seen. In 1988 RTM released malware for what has been perceived as academic discovery. Furthermore, the likelihood of RTM, as discussed, not foreseeing the impact the Morris worm would have on computers is plausible as the Morris worm was the largest ever malware incident of its time. In 1999 when Melissa was released, again the world heard an apology from the architect, as he testified that he was not aware of the damage Melissa would cause. However, by the time Melissa was released, the damage of malware was not new and had been demonstrated through previous malware outbreaks, such as the Morris worm. This suggests DLS should have foreseen the possibility of Melissa causing harm. In contrast, EMB demonstrated a clear intention to create malware to financially benefit himself and his colleagues, utilising botnets, which as demonstrated, are still a critical attribute of most cybercrimes. EMB’s botnet on take-down, being one of the largest botnets the world had ever seen (Graff, 2017).

Conclusion

The primary goal of this thesis was to determine whether rational choice theory could be used to explain the offending behaviour of malware architects. Specifically, it investigated whether the malware architects selected as case studies made rational choices to write and release malicious software. In answering this question, this thesis examined three malware architects across four decades, from the 1980s through to 2010 and onwards. This thesis then applied Cornish and Clarke's (2001) RCP's to each of malware architects as a way of determining whether rational choice was applicable.

When applying Cornish and Clarke's (2001) RCP's to each of the case studies, a distinct pattern emerged. With regards to the first three hypotheses of RCP, each case example fit each of the propositions outlined by Cornish and Clarke. However, questions arose regarding the last three propositions, as shall be demonstrated below.

RCP1 - *Crimes are purposive and deliberate acts, committed with the intention of benefitting the offender*.

Each case study demonstrated that the decision to write and distribute malware was deliberate, committed with purpose and with the intention of benefitting the offender in some way. RTM intended his program to monitor the growth of the internet, the results of which would have provided him with a substantial research advantage and prestige. DLS wrote Melissa for purely recreational purposes, and evidence suggests with the intention, of creating havoc only he could fix. EMB wrote malware specifically for financial gain.

RCP2 - ***In seeking to benefit themselves, offenders do not always succeed in making the best decisions because of the risks and uncertainty involved.***

As demonstrated in each of the case studies, the offenders did not always succeed in making the best decisions regarding the weighing up of risks and uncertainties. But that does not mean that the offenders were not aware of the risks involved in writing and distributing malware. Both RTM and DLS claimed their intentions were not malicious and that neither could foresee the consequences of their actions. As previously highlighted, the Morris worm was one of the first major malware incidents, so it is plausible that RTM did not foresee the dangers of releasing his worm. In contrast, DLS wrote Melissa when malware was already known to potentially cause havoc.

With regards to EMB, as demonstrated, his malware resulted in a different type of risk to that of RTM and DLS's malware. Whilst the previous malware caused critical failures to computer systems, EMB's malware was deigned to go undetected so that it could cause the most damage by covertly collecting information or using the computer for processing power. In fact, it appears the only risk EMB did not foresee was with regards to his original malware, ZeuS, being subject to piracy. This is an error EMB corrected in version two of ZeuS, GOZ, to which only EMB and his Business Club had access.

RCP3 - ***Offender decision making varies considerably with the nature of the crime.***

As per Cornish and Clarke's (1986) example of a burglar taking different precautions and measures depending on the neighbourhood and house they intend to burglar, each of the case studies appeared to have also taken different precautions and measures based on their motivations to offend. For example, RTM based his decision to offend on his search for knowledge. There is evidence to suggest he knew he was committing an offence as he took steps to conceal his identity (RTM having motive to hide his identity being a respected PhD student whose father held a high-level position working for the Government). However, when the malware did not follow the course RTM intended, he contacted his father, a man of some importance, ostensibly to confess, or at the very least, to gain some assistance. If RTM had not confessed to his father, RTM may never have been identified as the architect of the Morris worm.

DLS in comparison made malware 'for kicks'. As with RTM, DLS also took steps to conceal his identity, suggesting that he knew he was committing a crime. However, as he did not invest a lot of time into concealing his identity, it would appear DLS, as per his testimony, honestly believed Melissa would not spiral out of control. As such, when Melissa grew too fast too quickly he hid at his brother-in-law's house, suggesting that he feared he would be identified.

In contrast, EMB, a modern-day example of a malware architect, wrote ZeuS primarily for financial purposes. EMB then created a business around ZeuS and when that business began to fail, reinvented himself through a new malware with new procedures in place to ensure the same issues did not arise with version 2. It also appears EMB hid his identity very well as it took several years for his true identity to be discovered. When EMB's identity was discovered, he then acted on opportunities as they were presented to him, appearing to work with the Russian State in exchange for protection from prosecution.

As demonstrated, each of the three malware architects made a rational choice to commit crime through the distribution of malware. However, the examination of a modern day criminal offence, such as malware, challenges the traditional understandings of the last three of Cornish and Clarke's (2001) RCP, which focus on the involvement process and the event decision.

The involvement process is effected by background factors that influence an offender's decision to offend – the offenders are not making a spur of the moment decision. When examining RTM, a long-standing fascination with Unix security is evident, as demonstrated through both RTM's and RHM's involvement with the development of Unix and the identification of Unix security flaws. However, when examining DLS his motivation is not as evident. DLS demonstrated a 'good guy/bad guy' persona online, suggesting that he attained satisfaction out of creating problems that only he could fix. What is surprising about DLS was his age when he was apprehended, unlike RTM and EMB who were both in their very early twenties when they wrote their malware, DLS was 30. DLS was deeply imbedded in an online community when he was arrested, which may account for the apparent anomaly in his age. Another hypothesis is that the stereotypical cybercriminal no longer exists and expands over multiple age groups. A topic to be explored in another research paper.

With regards to EMB, the detailed biographical information identified in the previous two case studies is unavailable for this period prior to the discovery of ZeuS. However once ZeuS was discovered more and more information was published regarding EMB. It should be noted, that although it would be fascinating to identify the exact circumstances that initially drove EMB to write ZeuS, the answer to this question does not affect the results of this research paper.

As stated, once ZeuS was discovered information regarding EMB emerged from a multiplicity of sources indicating that he was a criminal entrepreneur who was financially motivated, as demonstrated through the criminal opportunities EMB made for himself. For example, the implementation of a private key system for executive buyers of ZeuS, the introduction of Cryptolocker and the reinvention of ZeuS to GOZ which only targeted frauds that were six and seven figures.

In regard to the criminal event, ‘traditional’ crime dictates that once an offender is ready to commit a crime the event decision begins. Then once the crime is committed and the criminal has escaped, the process of RCP is complete. However, as demonstrated, the idea of event decision is problematic for non-terrestrial crimes as they do not follow the same path outlined by RCP. Firstly, malware architects do not need to make an escape per se. Secondly, once the malware is released and the malware architects walks away from their computer, the criminal event does not end. Once malware is released it will continue to spread until every computer it has infected is either updated with effective anti-virus software or is deemed obsolete. In fact, ZeuS and variants of

ZeuS are still circulating today, twelve years after the event decision. This may be seen for a number of reasons, such as:

- Other criminals repurposing the malware for their own purposes;
- Victims not updating computers to the latest anti-virus and security protocols;
- Individuals not updating operating systems due to the cost of the upgrades or other reasons, such as being unsure how to operate the latest update, thereby operating end-of-line software that is no longer supported or protected by the manufacturer.

Based on the above findings, we may conclude that Cornish and Clarke's (2001) RCP's are partly outdated with regards to malware. As Yar (2005) explains, with regards to his investigation into routine activity theory and cybercrimes, there are distinct differences between terrestrial and non-terrestrial crimes, which may limit the application of criminological theory, supporting the proposition that cybercrime is indeed a new and distinct crime.

Regardless of the limitations of Cornish and Clarke's (1986) RCP's, the findings of this research suggest that each of the malware architects reviewed made a rational choice to release malware. As per Bachmann's (2008) research into hackers, malware architects reviewed in this dissertation demonstrated careful consideration, research, design and execution when writing and releasing their malware. As such, it may be concluded that each case study demonstrated complex and sophisticated judgements in their decision to offend, hence, each made a rational choice. For example, DLS may have written quite simple code in the form of Melissa, as opposed to RTM who wrote the Morris worm or EMB's ZeuS and GOZ. However, DLS had to consider how Melissa would be executed, who the target audience would be and how he could ensure Melissa would continue to spread once executed. This indicates a complex decision-making process.

It addition to the above findings, a number of other conclusions may be drawn from the research. Firstly, as demonstrated in the literature review and in the case study examples, early malware architects were computer hobbyists who were experimenting, as per RTM, or just "having fun", as per DLS (Bradbury, 2006). However, by 2003 this motivation had changed significantly to the malware architects' main intention typically being financial gain, as seen in the case example of EMB. Significantly, it appears that hobbyists, through demonstrating the potential of malware, inspired the financially motivated criminals seen from 2003 onwards. For example, the brute force mechanisms RTM created for password guessing are still used today. DLS, whilst not the first person to use email to spread a virus, was one of the first people to combine this with social engineering, so that his success caught the attention of the world.

The final case study examined, EMB, is a modern-day malware architect, who made a highly profitable business out of writing malicious code. Initially EMB rented out his botnet to other offenders during the 2000s, then in 2010 onwards it appears EMB created his own private malware, which utilised botnets, for his own benefit and that of his organisation. There is also evidence to suggest that EMB morphed one of his botnets into an intelligence gathering tool, however, it is the hypothesis of this paper, that this was purely a survival mechanism to ensure EMB continued to be protected by the Russian State.

Secondly, it appears that punishment at the hands of law was not considered by either RTM or DLS when they released their malware. This may be because neither assumed they would be identified as malware architects. As demonstrated in the analysis chapters, it is very rare for a malware architect to be identified, and if they are identified, to be arrested and prosecuted. Once EMB was identified as a malware architect by the FBI, it appears he formed a business relationship with the Russian State, providing a service in exchange for protection, rendering legislation and policy on malware propagation outside of Russia irrelevant. As stated, the Russian State has clearly indicated that only those Russian citizens committing cyber acts against Russians will be considered criminals and prosecuted.

Further, at the beginning of this research paper ten research questions were proposed, however two of these research questions were only addressed in the last case study:

4. Did the malware architect rely on trans-judicial boundaries to protect them from law enforcement when the malware was released, (that is, were they living in a separate country to where the malware was released)? and
7. If the malware architect has not been caught, what is their current status?

It is suggested that a useful avenue of further research would be to examine the role State protection plays in the implications of cyber-offences as well as the general implications of trans-judicial boundaries.

Lastly, Melissa and variants of Melissa were still propagating long after DLS had been arrested. Thus, a further avenue of research would be the responsibility of the malware architect towards their program and the damage it may be continuing to create, once they have been identified and arrested.

To conclude, criminology has been slow to fully grasp the dangers of cybercrime and malware. With the costs of cybercrime set to reach $2.1 trillion by 2019, the global impact of these offences is unprecedented. Therefore, further research is needed regarding the implications of cybercrime in general and the financial impact it is set to have. As demonstrated, rational choice theory may be seen as a useful theory when examining future research on malware architects. However, more research is needed as to the implications of these results. In addition, current legal sanctions imposed on malware architects did not appear to deter any of the architect's decisions to write malware. As such, additional research is needed to help formulate more useful deterrent policies specifically targeted at reducing malware distribution as well as malware victimisation.

Bibliography

Acohido, B. & Swartz, J., 2008. *Zero Day Threat.* Union Square Press.

Bachmann, M., 2008. *What makes them click? Applying the rational choice perspective to the hacking underground.* M.A. University of Mannheim.

Bachman, R. & Schutt, R. K., 2015. *Fundamentals of Research in Criminology and Criminal Justice.* 3rd ed. Sage.

Bossler, A. M. & Holt, T. J., 2011. Malware Victimisation: A Routine Activities Framework. In: K. Jaishankar, ed. *Cyber Criminology: Exploring Internet Crimes and Criminal Behavior.* CRC Press, pp. 317-343.

Bradbury, D., 2006. The Metaphorphosis of Malware Writers. *Computers and Security,* March, 25(2), pp. 89-91.

Broadhurst, R., 2006. Developments in the global law enforcement of cyber-crime. *Policing: An International Journal of Police Strategies & Management,,* Volume Vol.29(3), pp. 408-433.

Broadhurst, R., Grabosky, P., Alazab, M. & Chon, S., 2014. Organizations and Cyber crime: An Analysis of the Nature of Groups engaged in Cyber Crime. *International Journal of Cyber Criminology,* January-June, 8(1), pp. 1-20.

Clarke, Ronald V., & Cornish, Derek B. 1985. Modelling offenders' decisions: A framework of research and Policy. In M. Tonry & N. Morris (eds), *Crime and Justice: An annual review of research* (Vol 6). Chicago: University of Chicago Press.

Clarkson, Michael. *Beating the Superbug: Recent Developments in Worms and Viruses* SANs Institute, Reading Room. Available at: https://www.sans.org/reading-room/whitepapers/malicious/beating-superbug-developments-worms-viruses-146 [Accessed 4 October 2017]

Cluley, G., 2009. *Memories of the Melissa Virus.* [Online] Available at: https://nakedsecurity.sophos.com/2009/03/26/memories-melissa-virus/ [Accessed 3 July 2017].

Cluley, G., 2013. *25 years ago - the Morris worm struck the internet.* [Online] Available at: https://www.grahamcluley.com/morris-worm/ [Accessed 24 July 2017].

Cohen, F., 1984. *Fred Cohen & Associates.* [Online] Available at: http://all.net/books/virus/part5.html [Accessed 1 June 2017].

Computer Hope, 2017. *Is it a crime to make a computer virus?.* [Online] Available at: https://www.computerhope.com/issues/ch001196.htm [Accessed 21 August 2017].

Cornish, D. & Clarke, R., 1986. *The Reasoning Criminal: Rational Choice Perspectives of Offending.* Transaction Publishers.

Cornish, D. B. & Clarke, R. V., 1987a. Understanding criminal displacement: An application of rational choice theory. *Criminology,* 25(4), pp. 933-947.

Cornish, D. B. & Clarke, R. V., 2001. Rational Choice. In: R. Paternoster & R. Bachmann, eds. *Explaining Criminals and Crime: Essays in Contemporary Criminological Theory.* Los Angeles: Roxbury.

Cornish, D. & Clarke, R., 1987b. Introduction. In: D. Cornish & R. Clarke, eds. *The Reasoning Criminal: Rational Choice Perspectives on Offending.* Springer-Verlag, pp. 1-16.

Creswell, J. W. & Poth, C. N., 2018. *Qualitative Inquiry and Research Design: Choosing Among Five Approaches.* 4th ed. Samsung Hub: Sage.

Cross, S. B. D., 2014. *The "Hero Sydnrome".* [Online] Available at: http://www.cji.edu/site/assets/files/1921/the_hero_syndrome.pdf [Accessed 18 August 2017].

Cyberspace Law and Policy Centre, 2009. *Cyberspace Law and Policy Centre.* [online] http://www.cyberlawcentre.org [Accessed 18 August 2017]. darksheer, 2013. *PacketStorm-Exploits/9903-exploits/melissa.macro.virus.txt.* [Online] Available at: https://github.com/BuddhaLabs/PacketStorm-Exploits/blob/master/9903-exploits/melissa.macro.virus.txt [Accessed 27 September 2017].

Dellinger, A.J. 2016. *At the malware museum, a nostalgic gallery of old-school viruses*. [Online] Available at: https://www.dailydot.com/debug/malware-museum-internet-archive/ [Accessed 5 October 2017].

Denning, D., 2006. *Cyberwarriors.* [Online] Available at: http://hir.harvard.edu/article/?a=905 [Accessed 27 September 2017].

Drozhzhin, A., 2015. *Is your PC a part of a botnet? Check it!.* [Online] Available at: https://blog.kaspersky.com/simda-botnet-check/8304/ [Accessed 10 June 2017].

DuPaul, N., 2012. *Common Malware Types: Cybersecurity 101.* [Online] Available at: https://www.veracode.com/blog/2012/10/common-malware-types-cybersecurity-101 [Accessed 1 September 2017].

Editor, 2016. *Flashback Friday: The Melissa virus.* [Online] Available at: https://www.welivesecurity.com/2016/07/15/flashback-friday-melissa-virus/ [Accessed 28 July 2017].

Eichin, M. W. & Rochlis, J. A., 1989a. *Massachusetts Institute of Technology.* [Online] Available at: http://denninginstitute.com/modules/acmpkp/security/texts/INTWORM.PDF [Accessed 24 June 2017].

Eichin, M. W. & Rochlis, J. A., 1989b. *With Microscope and Tweezers.* [Online] Available at: http://web.mit.edu/user/e/i/eichin/www/virus/chronology.html [Accessed 24 June 2017].

Eisenberg, T. et al., 1989. *The Compuiter Worm: A Report to the Provost of Cornell University on an Investigation Conducted by The Commission of Preliminary Enquiry*. Cornell University.

Encyclopedia Britannica, 2017. *Y2K bug: Computer Science.* [Online] Available at: https://www.britannica.com/technology/Y2K-bug [Accessed 27 September 2017].

European Treaty Series - No. 185, 2001. *Convention on Cybercrime.* Budapest, 23.XI

Estes, A. C., 2017. *The World's Most Wanted Hacker Sounds Like A Goddamn James Bond Villain* [Online] Available at: https://www.gizmodo.com.au/2017/03/the-worlds-most-wanted-hacker-sounds-like-a-goddamn-james-bond-villain/ [Accessed 1 August 2017].

Federal Bureau of Investigations, 2017. *FBI Most Wanted.* [Online] Available at: https://www.fbi.gov/wanted/cyber/evgeniy-mikhailovich-bogachev [Accessed 1 August 2017].

Ferrell, J. & Ilan, J., 2013. Crime, culture, and everyday life. In: C. Hale, K. Hayware, A. Wahidin & E. Wincup, eds. *Criminology.* Third ed. Oxford.

Foley, M. J., 1999. *'Melissa' virus swamps corporate e-mail.* [Online] Available at: http://www.zdnet.com/article/melissa-virus-swamps-corporate-e-mail/ [Accessed 7 July 2017].

Furnell, S., 2012. Hackers, viruses and malicious software. In: J. Yvonne & Y. Majid, eds. *Handbook of Internet Crime.* Routledge, pp. 173-193.

Garber, L., 1999. Melissa Virus Creates a New Type of Threat. *Technology News,* June.pp. 16-19.

Gordon, S., 2000. *Virus Writers: The End of The Innocence?.* [Online] Available at: http://vxer.org/lib/asg12.html [Accessed 27 September 2017].

Gordon, S. & Ma, Q., 2003. *Convergence of Virus Writers and Hackers: Fact or Fantasy?* Symantec Corporation.

Grabosky, P., 2016. Cybercrime. Keynotes in criminological and crime justice series. Oxford University Press.

Grabosky, P. & Smith, R. G., 1998. *Crime in the Digital Age.* Transaction Publishers. The Federation Press.

Graff, G. M., 2017. *Inside the Hunt for Russia's Most Notorious Hacker.* [Online] Available at: https://www.wired.com/2017/03/russian-hacker-spy-botnet/ [Accessed 7 August 2017].

Graycar, A. & Grabosky, P. eds., 2002. *The Cambridge Handbook of Australian Criminology.* Cambridge University Press.

Grunwald, M., 1999. *Programmer Called Sire of 'Melissa' Virus.* [Online] Available at: http://www.washingtonpost.com/wpsrv/business/longterm/melissavirus/melissa040399.htm [Accessed 27 July 2017].

Hafner, K. & Markoff, J., 1995. *Cyberpunk: Outlaws and Hackers of the Computer Frontier.* Touchstone: Simon & Schuster.

Hayward, K. & Morrison, W., 2013. Theoretical criminology: a starting point. In: C. Hale, K. Hayware, A. Wahidin & E. Wincup, eds. *Criminology.* Third ed. Oxford, pp. 65-97.

Hinde, S., 2004. Hacking Gains Momentum. *Computer Fraud & Security,* November, 2004(11), pp. 13-15.

Holguin, J., 2002. *'Melissa' creator gets 2nd jail term.* [Online] Available at: www.cbsnews.com/news/melissa-creator-gets-2nd-jail-term [Accessed 7 June 2017].

Holt, T. J. & Bossler, A. M., 2016. *Cybercrime in Progress: Theory and Prevention of Technology-enabled Offenses.* Routledge.

Holt, T. J. & Schell, B., 2011. *Corporate hacking and technology-driven crime: Social dynamics and implications.* New York: Hershey.

House of Representatives Standing Committee on Communications, 2010. *Hackers, Fraudsters and Botnets: Tackling the Problem of Cyber Crime,* House of Representatives Committees.

Kigerl, A., 2012. Routine activity theory and the determinants of high cybercrime c ountries. *Social Science Computer Review,* 30(4), pp. 470-486.

Knight, W., 2003. *"Slammer" worm chokes the internet.* [Online] Available at: https://www.newscientist.com/article/dn3309-slammer-worm-chokes-the-internet/ [Accessed 21 July 2017].

Kocieniewski, D., 1999. *Melissa Virus Suspect Caught.* [Online] Available at: https://partners.nytimes.com/library/tech/99/04/biztech/articles/03melissa.html [Accessed 3 July 2017].

Krebs, B., 2012. *Zeus Trojan Author Ran With Spam Kingpins.* [Online] Available at: https://krebsonsecurity.com/2012/02/zeus-trojan-author-ran-with-spam-kingpins/ [Accessed 9 August 2017].

Krebs, B., 2014. *Operation Tovar' Targets 'Gameover' ZeuS Botnet, CryptoLocker Scourge.* [Online] Available at: https://krebsonsecurity.com/2014/06/operation-tovar-targets-gameover-zeus-botnet-cryptolocker-scourge/ [Accessed 7 August 2017].

Krebs, B., 2015. *FBI: $3M Bounty for ZeuS Trojan Author.* [Online] Available at: https://krebsonsecurity.com/2015/02/fbi-3m-bounty-for-zeus-trojan-author/#more-30070 [Accessed 7 August 2017].

Krebs, B., 2015. *Inside the $100M 'Business Club' Crime Gang.* [Online] Available at: https://krebsonsecurity.com/tag/evgeniy-mikhailovich-bogachev/ [Accessed 4 August 2017].

Krebs, B., 2015. *Krebs on security.* [Online] Available at: https://krebsonsecurity.com/tag/evgeniy-mikhailovich-bogachev/ [Accessed 26 May 2017].

Krebs, B., 2015. *Krebs on security.* [Online] Available at: https://krebsonsecurity.com/tag/evgeniy-mikhailovich-bogachev/ [Accessed 26 May 2017].

Lammers, W. J., 2004. Ch4 Developing Research Questions - Hypotheses and Variables. In: *Fundamentals of Behavioral Research.* Thomson/Wadsworth.

Left, S., 2001. *Code Red virus traced to China.* [Online] Available at: https://www.theguardian.com/technology/2001/aug/31/viruses.security [Accessed 21 July 2017].

Lemonnier, J., 2015. *What is Trojan horse malware?.* [Online] Available at: https://www.avg.com/en/signal/what-is-a-trojan [Accessed 27 September 2017].

Leyden, J., 2003. *Welsh virus writer Vallor jailed for two years.* [Online] Available at: https://www.theregister.co.uk/2003/01/21/welsh_virus_writer_vallor_jailed/ [Accessed 18 August 2017].

Malenkovich, S., 2013. *Morris Worm Turns 25.* [Online] Available at: https://www.kaspersky.com/blog/morris-worm-turns-25/3065/ [Accessed 27 September 2017].

Markoff, J., 1990. *Student Testifies His Error Jammed Computer Network.* [Online] Available at: http://www.nytimes.com/1990/01/19/us/student-testifies-his-error-jammed-computer-network.html?rref=collection%2Fbyline%2Fjohn-markoff&action=click&contentCollection=undefined&region=stream&module=

stream_unit&version=search&contentPlacement=4&pgtype=collection [Accessed 29 June 2017].

Marsan, C. D., 2008. *Morris worm turns 20: Look what it's done.* [Online] Available at: http://www.networkworld.com/article/2268919/lan-wan/morris-worm-turns-20--look-what-it-s-done.html [Accessed 27 July 2017].

McDonough, J. & McDonough, S., 1997. *Research Methods for English Language Teachers.* London: Arnold.

McNamara, P., 2009. *Melissa virus turning 10 ... (age of the stripper unknown).* [Online] Available at: http://www.networkworld.com/article/2235008/data-center/melissa-virus-turning-10------age-of-the-stripper-unknown-.html [Accessed 28 July 2017].

Messmer, E., 2008. *Tech Talk: Where'd it Come From, Anyway?.* [Online] Available at: http://www.pcworld.com/article/147698/tech.html [Accessed 10 August 2017].

Morgan, S., 2016. *Cyber crime costs projected to reach $2 trillion by 2019.* [Online] Available at: https://www.forbes.com/sites/stevemorgan/2016/01/17/cyber-crime-costs-projected-to-reach-2-trillion-by-2019 [Accessed 19 September 2017].

Morris, R. T., 1985. *A Weakness in the 4.2BSD Unix† TCP/IP Software,* s.l.: AT&T Bell Laboratories Murray Hill, New Jersey 07974.

Newman, G., 1997. Introduction: Towards a theory of situational crime prevention. In: *Rational choice and situational crime prevention.* Ashgate. Darthmore

Newman, G., Clarke, R. V. & Shoham, S. G. eds., 1997. *Rational Choice and Situational Crime Prevention.* Ashgate. Darthmore.

Noaks, L. & Wincup, E., 2004. *Criminological Research - Understanding Qualitative Methods.* London: Sage.

OfficeArticles, 2014. *About Normal.dot in Microsoft Word.* [Online] Available at: http://www.officearticles.com/word/about_normal_dot_in_microsoft_word.htm [Accessed 27 September 2017].

Panda Security, 2013. *The Most Famous Virus History: Melissa.A.* [Online] Available at: http://www.pandasecurity.com/mediacenter/malware/most-famous-virus-history-melissa/ [Accessed 20 June 2017].

Pucci, J., 2016. *Throwback Thursday: Cornell student Robert Morris guilty of unleashing first Internet worm.* [Online] Available at: http://www.syracuse.com/vintage/2016/01/throwback_thursday_cornell_stu.html [Accessed 24 June 2027].

Quora, 2016. *What does it take to become a computer architect?.* [Online] Available at: https://www.forbes.com/sites/quora/2016/10/06/what-does-it-take-to-become-a-acomputer-architect/?s=trending#5ce30d5b2c4f [Accessed 19 September 2017].

Radware, 2017. *DDoS Attack Definitions - DDoSPedia.* [Online] Available at: https://security.radware.com/ddos-knowledge-center/ddospedia/morris-worm/ [Accessed 27 September 2017].

Rouse, M., 2006a. *Sendmail.* [Online] Available at: http://whatis.techtarget.com/definition/sendmail [Accessed 18 August 2017].

Rouse, M., 2006b. *man in the browser.* [Online] Available at: http://searchsecurity.techtarget.com/definition/man-in-the-browser [Accessed 27 September 2017].

Rouse, M., 2007. *hacktivism.* [Online] Available at: http://searchsecurity.techtarget.com/definition/hacktivism [Accessed 6 October 2017].

Rouse, M., 2016a. *denial of service (DoS) attack.* [Online] Available at: http://searchsecurity.techtarget.com/definition/denial-of-service [Accessed 27 September 2017].

Rouse, M., 2016b. *Definition: Social Engineering.* [Online] Available at: http://searchsecurity.techtarget.com/definition/social-engineering [Accessed 3 July 2017].

Rouse, M., 2017a. *distributed denial of service (DDoS) attack.* [Online] Available at: http://searchsecurity.techtarget.com/definition/distributed-denial-of-service-attack [Accessed 6 October 2017].

Rouse, M., 2017b. *keylogger (keystroke logger or system monitor).* [Online] Available at: http://searchsecurity.techtarget.com/definition/keylogger [Accessed 6 October 2017].

Sandee, M., 2015. *GameOver ZeuS. Backgrounds on the Badguys and the Backends.* Fox-IT.

Sandywell, B., 2012. On the globalisation of crime: the Internet and the new criminality. In: Y. Jewkes & M. Yar, eds. *Handbook of Internet Crime.* Routledge.

Sartain, J., 2015. *5 essential tips for creating Excel macros.* [Online] Available at: https://www.pcworld.com/article/2880353/software-productivity/5-essential-tips-for-creating-excel-macros.html [Accessed 27 September 2017].

Schwirtz, M. & Goldstein, J., 2017. *Russian Espionage Piggybacks on a Cybercriminal's Hacking.* [Online] Available at: https://www.nytimes.com/2017/03/12/world/europe/russia-hacker-evgeniy-bogachev.html?mcubz=0 [Accessed 27 September 2017].

SecPoint, 2017. *What is a script kiddie?.* [Online] Available at: https://www.secpoint.com/what-is-a-script-kiddie.html [Accessed 13 September 2017].

Seeley, D., 1989. *A Tour of the Worm,* University of Utah.

Seltzer, L., 2013. *The Morris Worm: Internet malware turns 25.* [Online] Available at: http://www.zdnet.com/article/the-morris-worm-internet-malware-turns-25/ [Accessed 21 July 2017].

Spafford, E. H., 1989. The Internet Worm: Crisis and Aftermath. *Communications of the ACM,* June, 32(6), pp. 678-687.

Standler, R. B., 2002. *Melissa Virus.* [Online] Available at: http://www.rbs2.com/cvirus.htm#anchor222222 [Accessed 19 June 2017].

Stevenson, A., 2016. *The Russian government may be protectecting the creator of the world's most infamous malware.* [Online] Available at: https://www.businessinsider.com.au/gameover-zeus-alleged-author-may-be-getting-help-from-the-russian-government-2015-8?r=UK&IR=T [Accessed 1 August 2017].

Stone-Gross, B., 2012. *The Lifecycle of Peer to Peer (Gameover) ZeuS.* [Online] Available at: https://www.secureworks.com/research/the_lifecycle_of_peer_to_peer_gameover_zeus [Accessed 2017 August 2017].

Sullivan, B., 2004. *NBC News.* [Online] Available at: http://www.nbcnews.com/id/4946173/ns/technology_and_science-security/t/sasser-arrest-rare-victory-virus-wars/#.WXFa0sZL2Rs [Accessed 21 July 2017].

Swain, B., 2009. *What are malware, viruses, Spyware, and cookies, and what differentiates them?.* [Online] Available at: https://www.symantec.com/connect/articles/what-are-malware-viruses-spyware-and-cookies-and-what-differentiates-them [Accessed 14 May 2017].

Tavares, C., 1995. *Origins of the Cookie Monster.* [Online] Available at: http://www.multicians.org/cookie.html [Accessed 27 September 2017].

Taylor, P., 2001. Hacktivism: in search of lost ethics?. In: D. S. Wall, ed. *Crime and the Internet.* Routledge, pp. 59-73.

Taylor, P. A., 1999. *Hackers. Crime in the digital sublime.* Routledge.

Technopedia, 2017. *Technopedia.* [Online] Available at: https://www.techopedia.com/definition/27295/blaster-worm [Accessed 21 July 2017].

The Parliament of the Commonwealth of Australia, 2010. *Hackers, Fraudsters and Botnets: Tackling the Problem of Cyber Crime.* House of Represenatives. Standing Committee on Communications.

U.S. Department of Justice, 2002. *Creator of Melissa Computer Virus Sentenced to 20 Months in Federal Prison.* Newark(New Jersey): United States Department of Justice.

U.S. v. David Lee Smith (2002).

United States of America, Appellee, v. Robert Tappan Morris, Defendant-Appellant (1991).

United States v. Morris (1990).

Walker, J., 1985. *Computer Recreations: Letter to the Editor.* [Online] Available at: http://www.fourmilab.ch/documents/univac/animal.html [Accessed 27 September 2017].

Wall, D., ed., 2001. *Crime and the Internet.* Routledge.

Wall, D., 2001. *Crime and the Internet.* Routledge.

Wall, D., 2007. *Cybercrime: The Transformation of Crime in the Information Age.* Policy Press.

Wall, D. S. & Williams, M. L., 2013. Policing cybercrime: networked and social media technologies and the challenges for policing. *Policing and Society,* 26 March, 23(4), pp. 409-412.

Wall, D., 2012. Criminalising cyberspace: the rise of the Internet as a 'crime problem'. In: Y. Jewkes & M. Yar, eds. *Handbook of Internet Crime.* Routledge.

White, R. & Haines, F., 2001. *Crime and Criminology: An Introduction.* 2nd ed. Oxford.

Yar, Majid, 2005. The Novelty of 'Cybercrime'. An assessment in light of routine activity theory. In: *European Journal of Criminology*. 1 March, 2(4), pp. 407-427.

Yates, 2004. *Criminological Ethnography: Risks, Dilemmas and their Negotiation.* [Online] Available at: https://www.ncjrs.gov/pdffiles1/nij/Mesko/208043.pdf [Accessed 27 September 2017].

Yin, R. K., 1984. *Case Study Research: Design and Methods.* 2nd ed. Sage.

Zainal, Z., 2007. Case Study as a Research Method. *Jurnal Kemanusiaan,* Jun.Volume 9.

Zetter, K., 2009. *Nov. 10, 1983: Computer 'Virus' is born.* [Online] Available at: https://www.wired.com/2009/11/1110fred-cohen-first-computer-virus/ [Accessed 19 September 2017].

Zetter, K., 2014. *An Unprecedented Look at Stuxnet, the World’s First Digital Weapon.* [Online] [Accessed 17 May 2017].

Zorabedian, J., 2015. *Twin brothers accused of leading phishing gang busted by Russian police.* [Online] Available at:

https://nakedsecurity.sophos.com/2015/06/04/twin-brothers-accused-of-leading-phishing-gang-busted-by-russian-police/ [Accessed 1 August 2017].