2016

# Understanding the Impact of Hacker Innovation upon IS Security Countermeasures

Sean M. Zadig
*Nova Southeastern University*, seanzadig@gmail.com

# Understanding the Impact of Hacker Innovation upon IS Security Countermeasures

by

Sean M. Zadig

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in
Information Systems

College of Engineering and Computing
Nova Southeastern University

August 2016

We hereby certify that this dissertation, submitted Sean Zadig, conforms to acceptable standards and is fully adequate in scope and quality to fulfill the dissertation requirements for the degree of Doctor of Philosophy.

Gurivender P. Tejay Ph.D. — Date
Chairperson of Dissertation Committee

Ling Wang, Ph.D. — Date
Dissertation Committee Member

Stefan Savage, Ph.D. — Date
Dissertation Committee Member

Approved:

Yong X. Tao, Ph.D., P.E., FASME — Date
Dean, College of Engineering and Computing

College of Engineering and Computing
Nova Southeastern University

2016

## Abstract

Hackers external to the organization continue to wreak havoc upon the information systems infrastructure of firms through breaches of security defenses, despite constant development of and continual investment in new IS security countermeasures by security professionals and vendors. These breaches are exceedingly costly and damaging to the affected organizations. The continued success of hackers in the face of massive amounts of security investments suggests that the defenders are losing and that the hackers can innovate at a much faster pace.

Underground hacker communities have been shown to be an environment where attackers can learn new techniques and share tools pertaining to the defeat of IS security countermeasures. This research sought to understand the manner in which hackers diffuse innovations within these communities. Employing a multi-site, positivist case study approach of four separate hacking communities, the study examined how hackers develop, communicate, and eventually adopt these new techniques and tools, so as to better inform future attempts at mitigating these attacks. The research found that three classes of change agents are influential in the diffusion and adoption of an innovation: the developer/introducer of the innovation to the community, the senior member of a community, and the author of tutorials. Additionally, the research found that three innovation factors are key to successful diffusion and adoption: the compatibility of the innovation to the needs of the community, the complexity of the innovation, and the change in image conferred upon the member from adopting the innovation. The research also described the process by which innovations are adopted within the hacking communities and detailed phases in this process which are unique to these communities.

# Acknowledgements

Over the past seven years I have received a tremendous amount of support and encouragement from a number of friends, family, and scholars. I would first like to thank Dr. Gurvirender Tejay, my dissertation chair, guide, and friend. His mentorship – from my very first class at Nova Southeastern – has been a source of inspiration and example of true scholarship. I would also like to thank the remainder of my committee, Dr. Ling Wang and Dr. Stefan Savage, for their support and guidance with this study. Many of the students in my doctoral cohort also provided valuable advice throughout the process, including Drs. Dawn Dunkerley, Gary Doss, and Arlo Bess.

I would also like to thank my family, including my lovely and patient wife Heather for her encouragement to pursue this path, and my children, Fiona and Séamus. Thank you for dealing with the late nights, travel to conferences, and overall obsession with this project for the past few years.

# Table of Contents

**Chapters**

## List of Tables

## List of Figures

# Chapter 1

# Introduction

## 1.1 Background

Over the past two decades, the frequency and severity of criminal attacks upon Information Systems (IS) resources and assets have risen dramatically (Dowland, Furnell, Illingworth, & Reynolds, 1999; Verizon, 2010). As the world becomes increasingly reliant upon IS and the Internet for many important aspects of a functioning society – including entertainment, social connections, government programs, and commerce, as well as critical infrastructure components such as banking, power generation, and defense – disruptions to those aspects can have far-reaching financial and social consequences. Furthermore, organizations have embraced information technology as both a way to improve operating efficiency as well as an industry in its own right (Oliner & Sichel, 2000), and over time firms have so tightly integrated IS within their operations that it has indeed become the very core of business (Dhillon, 1995). As such, disruption of an organization's IS infrastructure by malicious attackers can have significant impacts upon the affected organization.

Organizations' IS have also increasingly been the target of attackers located outside the organization, which today outnumber the "insider" attacks originating from within the organization (Richardson, 2011). These external hackers have undergone a steady evolution from individuals motivated by challenge, curiosity, or notoriety (Hoath & Mulhall, 1998; Jordan & Taylor, 1998; Leeson & Coyne, 2005) to dedicated and well-organized computer criminals with a financial motivation rooted firmly in profit (Kshetri, 2006; Moore, Clayton, & Anderson, 2009). Attacks from external hackers have also claimed a greater financial impact upon the affected organizations, and most successful hacks often target economic-oriented organizations operating in the financial, hospitality, or retail industries (Verizon, 2010). As an example, the widely publicized attacks perpetrated against the retail firm TJX resulted in significant negative financial disruptions, including remediation costs, shareholder lawsuits, a loss of consumer confidence, and a reduction in market capitalization (Holt & Lampke, 2010; Xu, Grant, Nyugen, & Dai, 2008). Prior research has shown that the publicity of such attacks against an organization's IS can have severe financial effects upon the share price of the targeted firm (Cavusoglu, Mishra, & Raghunathan, 2004a) especially when the attack involves the breach of confidential information (Campbell, Gordon, Loeb, & Zhou, 2003).

Because of the possibility of negative financial impacts following a successful attack, firms have heavily invested in IS security countermeasures as a defense against attackers. According to the 2010/2011 Computer Security Institute survey, traditional IS security countermeasures are widely deployed – 97% of organizations utilize antivirus programs, 94.9% utilize firewalls, and 62.4% utilize intrusion detection systems, for example – although satisfaction with these countermeasures by organizational IS security officers hovers at around 40% (Richardson, 2011). Despite the presence of these countermeasures within organizations, users still continue to be victimized by attackers (Holt & Lampke, 2010). Furthermore, research into the deterrent value of these countermeasures has indicated that hackers are generally unfazed by the IS security countermeasures deployed by organizations and can generally find other methods of gaining access to targeted organizations (Tejay & Zadig, 2012), calling into question their efficacy.

Furthermore, Anderson et al. (2012) describe the economic state of IS security as being extremely lopsided, with defenders spending disproportionate amounts of money on countermeasures when compared with the illicit gains made by the attackers. According to Anderson (2001), even moderately-resourced attackers are able to defeat the security defenses of large and complex organizations, due to the sheer number of vulnerabilities that can be found and the necessity of defenders to secure everywhere in their infrastructure, a situation he calls a "Tragedy of the Commons in security". And yet, spending by organizations upon IS security continues to increase (Buzzard, 1999; Mercuri, 2003), even as the incidences of data loss from malicious attackers rise, along with the financial loss per successful attack (Richardson, 2011). Put succinctly, attackers and defenders are in an IS security arms race, where the brunt of the costs are born by the defenders, and the attackers are winning by almost every measure.

## 1.2 Problem Statement

To be able to consistently defeat the continual development and investment in sophisticated IS security countermeasures by organizations, and to ensure continued success in their criminal activities as described above, hackers must need to constantly develop new methods of breaching the defenses of organizations. Hackers are extremely collaborative and routinely share knowledge within specialized communities, in order to develop innovative approaches to circumvent security controls (Bachmann, 2010; Holt

2009; Holt, Strumsky, Smirnova, & Kilger, 2012; Jordan & Taylor, 1998). In particular, Bratus (2007) identifies *attack techniques* and *attack tool development* as areas of active learning within hacker communities.

Holt (2007) identified web forums as a type of community where hackers can share knowledge and tutorials regarding various aspects of hacking, in a supportive and reinforcing environment. Bratus (2007) also described hacker magazines and hacker conferences as other locations where "black hat" attackers can learn new hacking skills from other hackers. Jordan and Taylor (1998) examined both internal and external factors involved in the operation of underground hacker communities, although their research was done during the time of hacking for challenge or notoriety and before financial aspects became prime motivators of computer attacks. Despite these examinations, however, there remains a general lack of understanding of how new hacking techniques are developed within these communities, and how the communication of these new ideas may play a role in the eventual adoption of the innovations by hackers.

The research problem of this study, therefore, is to understand the nature and characteristics by which participants of hacking communities develop and adopt techniques to circumvent IS security controls. This research focuses specifically upon the areas of attack techniques and tool development, as identified by Bratus (2007). This research argues that the development and dispersion of innovations within hacker communities is a coordinated and structured process, akin to the innovation process that may be found within legitimate organizations. Far from being a collection of lone hackers who are committing attacks upon random targets, these communities are populated with well-organized attackers cooperating to engage in ever-more sophisticated and complex attacks upon victim organizations (Anderson et al. 2012; Moore, Clayton, & Anderson, 2009; Ollman, 2008; Thomas & Martin, 2006). An understanding of the innovative processes of attack technique and hacking tool development within these communities, therefore, is a crucial requirement for future efforts to reduce the effectiveness of hackers, and increase the IS security of organizations. This study is significant because it examines the previously-unexplored area of hacking technique development within hacker communities and the effect that this constant innovation has upon attacks directed towards organizations. By examining these factors of hacker effectiveness, researchers will be able to gain a holistic understanding of hacker attacks, and can seek to obtain additional methods for the mitigation of such attacks.

### 1.3 Research Questions and Propositions

Based upon the above research problem and argument, the following two research questions are appropriate:

1. *What characteristics affect the development and communication of emergent hacking techniques and tools within hacker communities?*
2. *How do the participants in these communities decide to adopt these techniques and employ them in attacks against organizations?*

The first research question deals with how hackers in these collaborative communities come to develop their new hacking techniques. Prior adoption research has shown that innovation occurs within an organization when a new product or process invention is introduced in response to a need or want (Utterback, 1971), and as such an examination of these needs will be illuminative for the present study. Once the innovation is introduced to the hacking community, how the innovation is communicated within the community is also of relevance to this question. Ebadi and Utterback (1984) had found that a number of factors, to include the frequency of communication, position of the innovation within the organization, and diversity of parties in communication positively affect the success of an innovation; overly formalized communication, however, was shown to have a negative effect. Therefore, how hackers communicate regarding the new innovation will likely affect the eventual adoption of the new hacking techniques and tools and was an area requiring study.

As a result of this discussion of this first research question, two related propositions can be derived:

*P1: Participants of hacking communities develop new techniques when existing hacking techniques cease to reliably breach targeted organizations.*

*P2: The methods by which the new hacking techniques are communicated amongst the community affect the adoption of the innovations.*

The second research question deals with how new techniques are adopted once a new hacking technique is developed and communicated. For example, an examination of the characteristics of individual hackers who advocate for adoption of the innovation by the community will be necessary; Rogers (2003) describes these individuals as "change agents". Moore and Benbasat (1991), from seminal technology acceptance literature, describe a number of factors that are related to the adoption of new technologies; these factors will be evaluated in light of hacking innovation adoption within hacker communities. Also of relevance to this research is the determination of not only how an innovation is initially adopted, but whether or not the innovation continues to be used by the attackers, which may determine how it will be employed against victim organizations and their IS security countermeasures.

Similarly, the following propositions can be derived from this discussion:

*P3: Change agents within the hacking community are instrumental in the adoption of the hacking innovation.*

*P4: Continued use of a hacking innovation by community members will determine the ultimate adoption of an innovation.*

These research questions, and derived propositions, guided the present research study.

### 1.4 Relevance and Significance

As described above, the ability of hackers to consistently overcome advances in IS security countermeasures has been extremely problematic for organizations who wish to defend against these attacks. While prior research has shown that hackers participate in specialized communities where they can learn new hacking techniques (Bratus, 2007; Holt, 2009), in general the extant research has not examined the methods by which this innovation occurs. A greater understanding of the methods by which hackers develop new hacking techniques and tools may result in the development of enhanced IS security defenses that can be deployed by organizations, or in the ability by law enforcement or IS security firms to disrupt hacker communities by interrupting the innovation process or targeting key members of the community for arrest and prosecution.

Previous attempts at understanding hacker communities have focused upon in-person hacker conferences (Bratus, 2007) or, in the case of online forums, have been limited to either the examination of social pressures or subcultural norms present in these communities (Holt, 2007; Holt & Copes, 2010; Jordan & Taylor, 1998) or an analysis of the structure of the community itself (Lu, Polgar, Luo, & Cao, 2010). These attempts, while useful for understanding the mindset or organization of hackers, have not attempted to solve the problem of the constant attacks that organizations face, and as a result the costly cycle of attack and defend has continued unabated. The present study, on the other hand, sought to obtain an understanding of the process by which hackers develop and adopt new hacking techniques or hacking tools, which are subsequently used upon victim organizations. A detailed study of this process may open up new research horizons into the areas of hacker innovation and may lead to the creation of novel defenses that reflect this innovation process.

This study also added to the body of knowledge by conducting IS innovation research in a criminal context; prior IS innovation research has focused upon the legitimate development of IS within organizations. In the non-IS innovation realm, only a handful of studies have focused explicitly on criminal innovations, and these have emphasized white collar financial crimes – notably fraud in the subprime mortgage market (Koller, 2010), resource investor fraud (Baker & Faulkner, 2003), stock option fraud (Snyder, Priem, & Levitas, 2009), and the diffusion of check forgery innovations (Lacoste & Tremblay, 2003). This study, therefore, offers a new area of innovation research for IS scholars – that of the development of deliberately malicious IS innovations. This research also contributed to innovation research by adding to the relatively minor amount of research on criminal innovations in general.

Furthermore, these hacker communities are often invitation-only or require that participants prove their “street cred”, or ability to conduct criminal activity, before being admitted (Leydon, 2011). Hackers reside within specialized subcultures with their own unique speech, norms, and values (Holt, 2007; Holt, 2009) and outsiders can often have difficulty when they try to understand the inhabitants of those subcultures (Conti, 2000). Research into criminal organizations is itself a difficult issue within the criminology discipline (Feenan, 2002; Ferral & Hamm, 1998). As a result, only a few security researchers have attempted to study malicious (or "blackhat") attackers directly – instead, many have opted to investigate hackers who test IS security systems without explicitly breaking the law ("greyhats"), or substitute non-malicious users ("whitehats") or even university students in empirical studies of misuse (Mahmood, Siponen, Straub, Rao, & Raghu, 2010). This study, therefore, was a significant and novel research endeavor.

### 1.7 Scope and Definitions

A proper understanding of the key concepts and terms essential to this study is required before proceeding further. Leibenau and Backhouse (1990) describe *information* as a collection of data, arranged in some meaningful way and for some perceived purpose. They further describe an *information system* (IS) as the aggregate of information handling activities within the technical, formal, and informal levels of an organization. Given this definition, it is difficult to delineate where the organization ends and IS begins, and as such the organization and the information system are essentially indistinguishable (Dhillon, 1995). The formal level of IS encompasses organizational bureaucracy that governs rules of interaction, both inter-organizational and intra-organizational, while the informal level encompasses organizational subculture where the associated meanings, intentions, and beliefs are defined (Dhillon & Backhouse, 1994).

The technical level includes the organization's information technology components and enables and automates the functioning of the formal system. These three levels can therefore be viewed as a series of concentric rings, with the technical system in the middle, surrounded by the formal system of bureaucratic rules, which is in turn surrounded by the informal system that interprets and gives meaning (Dhillon & Backhouse, 1996).

Historically, classical studies of computer security have focused upon technical aspects and have emphasized solutions within the "CIA" sphere: confidentiality, integrity, and availability (Anderson, 2003). However, this definition ignores the organizational aspects of security. Expanding upon the definition of an IS described above, Dhillon (1995) describes *information systems security* as the minimization of risks that arise because of inconsistent and incoherent activities with respect to an organization's information handling activities. Anderson (2003) provides an alternate definition by defining IS security as "a well-informed sense of assurance that information risks and controls are in balance". This effort to minimize IS risk has been examined through many different lenses, including the evolution of IS security analysis and design (Baskerville, 1993), development of appropriate security policies (Baskerville & Siponen, 2002), and importantly for a discussion pertaining to computer crime, deterring computer abuse (Straub & Welke, 1998).

*Computer crime* (or alternatively, "*cybercrime*") has been defined in various ways. Doss (2012) states that the terms computer crime and cybercrime have different meanings, although they are often confused in the literature. Doss describes "cybercrime" as a crime that occurs utilizing a computer or information system, but it does not have the computer as the focus of the crime and are instead used as aids to the crime, such as in Internet-based fraud. Computer crimes, alternately, are crimes where the computer or information system is the focus of the crime, such as computer hacking, denial of service, or software piracy (Doss, 2012). For the purposes of this research, the term computer crime will be utilized in lieu of cybercrime to alleviate any confusion within the literature.

In line with this notion of computer crime, Kshertri (2005) defines a computer crime as "any crime that employs a computer network in any phase of the crime", while Chung, Chen, Chang, and Chou (2006) define it as "illegal computer-mediated activities that often take place in the global electronic networks". Both of these definitions focus upon the involvement of a computer network. Alternatively, the US Department of Justice (2010) provides a definition which does not include a networked component: that a computer crime involves either an individual who exceeds authorized access to a computer, which encompasses *insiders*, who have some form of authorized access to an organization's IS (D'Arcy & Hovav, 2009); or an individual who intrudes into a computer without authorization, which encompasses *external hackers*, who have no authorized access to an organization's IS. This research is specifically concerned with individuals located outside of the organization that are engaged in attacks over computer networks.

Attacks from hackers external to the organization are today more prevalent and more costly than insider attacks, as described previously. The external hackers that are to be examined in the present study are known as "blackhats", or those who conduct attacks for personal gain without concern for the damage that may be caused. Other hacker types –

including the “whitehats”, the hackers that do not conduct malicious attacks out of ethical considerations, or “greyhats”, the hackers that may conduct malicious attacks but who generally seek to improve security or warn users (Bratus, 2007; Holt, 2009) – are not the focus of this study. Blackhat hackers tend to believe they have much more to gain than to lose from conducting attacks and generally perceive a low likelihood of being identified and prosecuted for their criminal acts (Young, Zhang, & Prybutok, 2007). This consideration of risks versus reward implies that hackers are rational actors who make economic decisions, which has led to research employing game theory to model the interactions between external hackers and IS security defenders (Cavusoglu, Cavusoglu, & Raghunathan, 2004). In general, however, the IS security literature has shown a distinct preference towards an understanding of whitehat defenders, to the exclusion of the blackhat attackers (Mahmood, Siponen, Straub, Rao, & Raghu, 2010).

Organizations employ IS security *countermeasures*, or controls, as a means of defending against the constant barrage of hacker attacks. Straub (1990) described two classes of IS security countermeasures – deterrents, which may discourage potential offenders, and preventives, which impede potential attackers. Ransbotham and Mitra (2009) examined organizational use of IS security countermeasures in their study of paths to IS security compromise and found that these countermeasures can be described in five categories: access control, or restricting access to IS assets based upon need; vulnerability control, or the removing of known errors in hardware or software that can be exploited by attackers; feature control, or the adjustment of IS asset settings to reduce the potential for abuse; traffic control, or the monitoring and blocking of malicious traffic; and audit control, the documenting of systems that can be used for auditing.

Hackers can often be found in underground *hacker communities*, where they can obtain support and engage in the sharing of knowledge related to attacks (Holt, 2013; Jordan & Taylor, 1998). These communities typically take the form of web forums, newsgroups, and Internet Relay Chat (IRC) channels, and often include the use of jargon and slang unique to the underground hacker scene (Holt, 2009). Hacker communities often resemble marketplaces, with various hackers specializing in providing different types of criminal services for a fee (Holt, 2013). While hackers can also exchange information in off-line environments, such as hacker conferences (Bachmann, 2010; Holt 2009), the present study will focus upon online hacker communities. These hacker communities are meritocracies, where participants are evaluated based upon their technical skills and abilities, and by sharing hacking knowledge and techniques a hacker is able to demonstrate his proficiency and highlight his value to the community (Holt et al., 2012). The emphasis upon education is important, as these communities are primarily filled with hackers of lower technical skill and ability, who are trying to learn how to be more effective at conducting attacks, and with only a small number of truly "elite" hackers capable of providing such knowledge (Holt, 2007; Holt et al., 2012; Jordan & Taylor, 1998). Hackers in these forums frequently share tools and post fraudulently-obtained information for others to utilize (Holt & Lampke, 2010, Peretti, 2009). These forums often begin as a "swarm" structure with no obvious leaders as hackers meet and trade information, but evolve into more formalized "hub" structure as the community matures, with greater levels of organization, and frequently involve transnational constituents (UN Office on Drugs and Crime, 2013).

It is important to note that the areas of cyberwarfare and cyberespionage, (Everett, 2009, Goel, 2011, Kshetri, 2005, Mandiant, 2013; Richardson 2011) as well as cyberterrorism (Foltz, 2002; Hansen, Lowry, Meservy, & McDonald, 2007), even though they involve individuals that meet the definition of a "hacker", are outside of the scope of this paper, which is focused upon the diffusion of information within hacker communities of a purely criminal nature.

### 1.8 Summary

Organizations continue to suffer costly attacks from external hackers, even as spending on security countermeasures increases dramatically.   These attacks often have devastating consequences, to include reductions in share price, loss of customer confidence, and expensive remediation efforts.  Research has shown that hackers are collaborative and share knowledge in underground communities; to continually defeat organizational IS security countermeasures, these hackers must be constantly innovative in their attacks.  Today's hackers are increasingly motivated by financial goals and conduct attacks for profit, instead of the notoriety or challenge that motivated previous generations of hackers.

This study focuses upon the highly organized process of development and adoption of new hacking methods to overcome these countermeasures.  These communities of hackers in fact resemble legitimate organizations with their collaborative and innovative behavior, and reap the rewards when their successful hacking innovations allow them to penetrate organizations.

# Chapter 2

# Review of the Literature

## 2.1 Introduction

This chapter describes prior literature of relevance to an inquiry related to the diffusion of innovations pertaining to IS security countermeasure circumvention and defeat by external hackers. This literature includes an overview of research pertaining to hackers, computer crime, and the effects of hacker attacks upon the economics of information systems security, as well as research related to IS security countermeasures.

## 2.2 Hackers and Computer Criminals

The word "hacker" was originally a positive term when first used to describe individuals who were especially skilled at developing creative computer programs and algorithms (Bacchman, 2010). Individuals at the Massachusetts Institute of Technology first identified themselves as hackers in the 1960s and 1970s, when the computer use began to become more prevalent (Hoath & Mulhall, 1998; Turgeman-Goldschmid, 2005). These first hackers emphasized solving problems and building innovative software, and encouraged such traits as hacker cooperation, helping other hackers, and learning new methodologies, and such this definition of hacker can still be found in use in open-source development communities (Stewart & Gosain, 2006). Later, the term evolved to encompass computer criminals, but these early illicit hackers possessed a strong ethical code, usually informal but sometimes formalized in hacker publications, such as *2600* or

*Hacktivismo* (Gordon & Ma, 2003). Hackers initially began to cross into criminal behavior with the behavior known as “phreaking”, or breaking into telephone systems to make free long distance calls (Leeson & Coyne, 2005). Many of these earlier hackers fought to make protected information freely accessible in the hopes that a more egalitarian society would develop (Hollinger, 1991). Over time, however, the term has been negatively associated with computer criminals that often operate without such an ethical code, and who may engage in such activities purely for profit (Galbreth & Shor, 2010). While some utilize the term “crackers” for such malicious hackers (Barber, 2001a), for the most part the term “hacker” has become a negative one used to describe cybercriminals (Hollinger, 1991).

Historically, the hacker community has been male-dominated (Jordan & Taylor, 1998) and hackers are often college-aged (Xu, Hu, & Zhang, 2013) and come from upper-middle class backgrounds (Hollinger, 1991). Before the present age of financially-motivated attackers, most individuals who identified as hackers were motivated by sociological forces – including desires for challenge (Hoath & Mulhall, 1998), notoriety (Leeson & Coyne, 2005), or excitement and curiosity (Turgeman-Goldschmid, 2005). Many criminal hackers often describe themselves as being troublemakers when younger, whose deviant behavior evolved into hacking (Turgeman-Goldschmid, 2008). Hu, Zhang, and Xu (2012) explored this notion of early deviance and antisocialism and found that young people who possessed strong moral beliefs and exerted self-control, and who favored social activities such as sports over antisocial activities such as computer games, were much less likely to become hackers.

Despite the above, IS security researchers have shown a historical preference for focusing upon insider crime – that is, attacks upon IS that originated from within the organization by rogue employees or others with insider access (Schultz, 2002). After all, insiders have significant opportunities to engage in computer crimes due to their legitimate access to IS resources (Willison & Backhouse, 2006). Doss (2012) also points out that external attackers can become insiders through the use of malware, perhaps of a variety that provides covert access to internal resources. As such, a brief description of insider IS crime is warranted.

Straub and Nance (1990) examined the discovery and managerial discipline of computer abusers within organizations, and defined computer abuse as "unauthorized, deliberate, and internally recognized misuse of assets of the local organizational information system". Straub (1990) similarly focused primarily on insider attacks in a study involving the deterrent values of IS security countermeasures, and found that well-communicated security policies combined with the employment of security surveillance tools can result in significant reductions in internal abuse. D'Arcy and Hovav (2007) examined the deterrent effects of technical countermeasures (computer monitoring and preventive security software) and non-technical (security policies and security awareness programs) on internal abuse and found the non-technical countermeasures were more effective, and that one of the technical countermeasures – computer monitoring – did not have a significant countermeasure effect at all. In a follow-up study, D'Arcy and Hovav (2009) found that technically-savvy users are less impacted by some countermeasures, as they may feel like they can more easily overcome these controls.

Dhillon and Moores (2001) conducted two case studies of particularly egregious insider attacks involving the securities and banking industries, highlighting lax internal controls at the technical, formal, and informal levels. Willison and Backhouse (2006) examined the factors involved in opportunistic attacks upon IS by internal abusers, and suggested that the considering the offender's perspective may be useful in understanding attacks perpetrated by insiders against vulnerable IS resources. These studies, however, and others that focus upon insider attacks (Lee & Lee, 2002; Theoharidou, Kokolakis, Karyda, & Kiountouzis, 2005; Willison, 2000) often rely upon deterrent models that assume that organizations have access to offenders to administer sanctions, which typically does not apply in the case of external hackers. Organizations are often very proficient at informing internal users of the presence of security countermeasures and the severity of punishments for abuse (D'Arcy, Hovav, & Galletta, 2009), but extending that deterrence to external hackers is much more difficult.

Deterring internal hackers has been a significant area of research, and has focused upon the employment of criminological theories within the IS realm (Xu, Hu, & Zhang, 2013). Straub (1990) utilized General Deterrence Theory (GDT) to understand IS abuse, an approach that relies upon the administration of sanctions in response to criminal behavior as a means of deterrence. Strong and visible policing, combined with the promise of swift and harsh punishment, dissuades potential criminals from committing crimes by emphasizing that the risks of detection are too high and that the punishment if detected would be too severe to make it worth committing the crime in the first place. GDT relies upon the thinking that attackers engage in attacks based upon the maximization of benefit and the minimization of costs (Theoharidou, Kokolakis, Karyda,

& Kiountouzis, 2005). In an organizational context, Straub and Welke (1998), in their “Security Action Cycle”, state that potential IS abusers are deterred by robust security policies, effective detection mechanisms, and appropriate punishments.

Lee and Lee (2002) also examined Social Bond Theory (SBT) as a means of deterrence, which states that strong social bonds deter organizational insiders from engaging in criminal activity, as well as Social Learning Theory (SLT), which posits that individuals commit crime because they associate with delinquents and take on their values. In an IS context, employees who are more socially attached to their coworkers and who do not associate with computer criminals are less likely to engage in computer crime. Lee and Lee (2002) suggested a holistic model employing SBT, SLT, and GDT together to understand and deter insider IS abuse. Other research has examined the impacts of offender rationalization, motivation, and disgruntlement upon computer crime and deterrence (Willison & Warkentin, 2013).

Theories that focus upon the computer crime itself, instead of the computer criminal, have also been examined. One such theory is that of Situational Crime Prevention (SCP), which seeks to increase the risks of detection for criminal acts while reducing their rewards (Willison, 2000). Examples of deterrence through SCP include the hardening of potential targets through increased countermeasures, as well as the employment of surveillance. Other attempts at understanding the crime have included examinations of the choices that criminals make when conducting the criminal act, environmental factors that affect a criminal’s choice of a target or site, and the routine activities of the targets and how they interact with the tools employed by criminals (Willison, 2006). While these studies were intended for the insider threat, many of these theories may have applicability when considering attacks from outsiders as well (Zadig & Tejay, 2010).

## 2.3 Computer Crime External to the Organization

Computer crime is a rather broad field – after all, criminals are very adept at exploiting many aspects of IS for illicit gain. Of relevance to this study, Richardson (2011) identified malware – or *mal*icious soft*ware* – as a major problem for organizations that frequently contributes to the compromise of IS resources by external attackers. Computer viruses were one of the first types of malware, and initially rose to prominence in the 1980's (Cohen, 1987). These early computer viruses spread slowly at first, as many computers during this time period were not networked and instead relied upon infections that spread using floppy disks (Highland, 1997). An early exception to this was the Morris Worm, which infected 5% of the networked computers on the early Internet in 1988 (Orman, 2003). Later Internet-based viruses wrought increasingly expensive destruction, including the Melissa virus in 1999 and the ILOVEYOU virus in 2000, each of which cost millions or billions of dollars in damages to organizations worldwide (Bishop, 2000; Garber, 1999).

The landscape malware of today, however, is significantly more advanced and crowded. One of the most prevalent types of malware in existence today is known as a botnet, or a ro*bot net*work. Botnets consist of many infected victim computers known as bots that are controlled through a command and control infrastructure operated by computer hackers (Ianelli & Hackworth, 2006; UN Office on Drugs and Crime, 2013). Botnets are believed to infect many millions of computers worldwide (Weber, 2007), resulting in significant computing and bandwidth resources in the hands of criminals. According to Zadig and Tejay (2011), botnets are operated by hackers for a number of criminal purposes, including the sending of spam, financial theft from victim computers or online bank accounts, denial of service attacks, fraudulent clicks upon online advertisements, and information stealing, among others. The primary purpose for botnets is the generation of profits for the botmasters, as the hackers who control them are called, and it seems that profit has allowed these hackers to become more specialized and professional. For example, Ollman (2008) described how some computer-based hacking groups emulate legitimate IS firms, and employ large malware development teams, offer service level agreements and product support for customers, and provide managed criminal services. Moore, Clayton, and Anderson (2009) state that some of these groups even have dedicated budgets for research and development and testing, with the end result that "online crime has become organized and industrialized like no other crime, with the possible exception of the drug trade" (Moore et al., 2009, p. 17).

Indeed, economic thinking appears to be at the heart of modern hacker activities. One type of malware known as "fake antivirus" programs, designed to socially-engineer users into paying for fraudulent security programs, is estimated to have been purchased by over one million victims (Provos, Rajab, & Mavrommatis, 2009). Thomas and Martin (2006) examined underground computer crime marketplaces devoted to the trading of financial data and detailed significant amounts of specialization, where attackers can take on many roles involved in the compromise, sale, purchase, and liquidation of online bank accounts. Other criminal groups have ventured into the profitable market of "bulletproof hosting", offering hosting services where hackers can host criminal IS infrastructure without fear of removal due to abuse complaints or requests from law enforcement (Stone-Gross, Kruegel, Almeroth, Moser, & Kirda, 2009). And Choo (2008) states that many hacking groups have begun to employ various organizational structures to properly manage their criminal activities, while traditional organized crime groups have begun to expand their money-making operations online (Choo & Smith, 2008). According to the United Nations, approximately 80-90% of all computer crime now originates from some form of organized activity, be it highly organized groups or more loosely-organized criminal markets (UN Office on Drugs and Crime, 2013).

Hackers can also engage in significant attacks without the use of sophisticated or customized malware. For example, the "Anonymous" hackers employed rudimentary distributed denial of service (DDoS) tools that were nevertheless able to take numerous targeted organizations offline (Lua & Yow, 2011). Hackers also often employ commercial or open-source off-the-shelf vulnerability scanners to scan target networks for vulnerable applications or security holes that they can exploit to gain access (Barber, 2001b), or utilize "sniffer" software to intercept passwords or other sensitive information as it is transmitted over the network (Venter & Eloff, 2003). These tools are designed to overcome the IS security countermeasures deployed by organizations as defenses against hackers.

### 2.4 IS Security Countermeasures

Using the US Department of Justice (2010) definition above, computer crime originating from outside the organization occurs when an individual accesses a computer system without authorization. In order to access this system, the hacker must have bypassed or defeated the IS security countermeasures erected as a defense by the organization. Straub (1990) describes IS security countermeasures as both technical and non-technical methods to prevent and deter computer abuse, which fall into the classes of *deterrents*, such as security policies, and *preventives*, such as physical and technical security measures. Cavusoglu, Mishra, and Raghunathan (2004b) took a non-deterrence approach and divided countermeasures into categories of preventives and *detectives*, the latter being those that can detect and inform about security events without physically impeding them. As described previously, Ransbotham and Mitra (2009) detailed the types of technical countermeasures available to organizations in their work on countermeasures and the effects upon IS security compromises. These countermeasures can be classified into the broad categories of access control, vulnerability control, feature control, traffic control, and audit control. Tejay and Zadig (2012) provided a summary of these countermeasures, which can be found in Table 1.

**Table 1: IS Security Countermeasures** (from Tejay & Zadig, 2012)

| **Countermeasure** | **Definition** | **Examples** |
|---|---|---|
| Access Control | Restricting access by people and software based on need | Access control lists, user authentication |
| Vulnerability Control | Removing known errors in hardware and software that can be exploited for inappropriate use | Software patching, validation of user input |
| Feature Control | Setting parameters in devices and software to reduce inappropriate use | Disabling insecure or unused ports or protocols |
| Traffic Control | Monitoring and blocking traffic based on identification of inappropriate activity | Packet filtering, blocking traffic |
| Audit Control | Documentation of systems and activity that can be used for audits and actions | Logging and management of activity records |

Tejay and Zadig (2012) also developed a framework of IS security countermeasures (or “controls”) available to organizations in their work on the deterrent efficacy of these countermeasures. Figure 1 describes this framework and provides examples of where typical countermeasures would be located within the framework, which was based on the countermeasure categorization provided by Ransbotham and Mitra (2009). In this framework, countermeasures are broken into four quadrants, representing the domain of the particular countermeasure, either internal or external to the organization; and the nature of the countermeasure, either active or passive. As can be seen, these countermeasures focus primarily upon technical aspects of IS security, but also include the non-technical countermeasures within the legal system, which was not originally included by Ransbotham and Mitra.

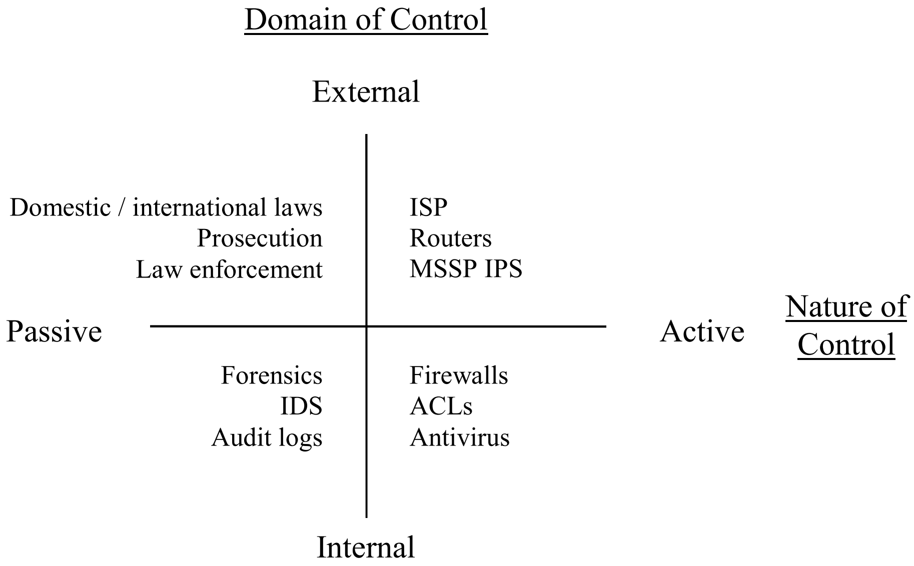


**Figure 1: IS Security Countermeasure Framework** (from Tejay & Zadig, 2012)

Internal active countermeasures consist of many of the standard technical controls employed by IS security professionals and described extensively within the literature. These countermeasures are considered *active* because they directly inhibit an attack from occurring (Tejay & Zadig, 2012), such as in the case of firewalls or access control lists which block non-conforming network connections or unauthorized users (Bishop, 2003; NIST, 1995) and antivirus software which physically deletes or quarantines files containing viruses (Venter & Eloff, 2003). These countermeasures are mostly those referred to as preventives by Cavusoglu, Mishra, and Raghunathan (2004b). Active countermeasures encompass the categories of access control, vulnerability control, feature control, and traffic control (with the exception of traffic monitoring) from Ransbotham and Mitra (2009), as each of these categories can be considered a direct impediment to attack. They are considered *internal* because they are physically located within the organization itself and are deployed within the organization's IS security infrastructure (Tejay & Zadig, 2012). Countermeasures of the internal active category are among the ones most regularly employed by IS security units (Richardson, 2011).

Organizations also frequently have external active countermeasures at their disposal. These controls are *external* because they are located outside of the organization, and may be under the control of external entities such as Internet Service Providers or Managed Security Service Providers (Tejay & Zadig, 2012). Examples of these controls include intrusion prevention systems (IPS) outside of the organization's firewall (Patel, Qassim, & Wills, 2010) or ISP-level defenses such as router access control lists or denial of service prevention devices (Mirkovic & Reiher, 2004). Richardson (2011) states that controls of this type, such as IPS, are used by roughly half of IS security departments.

Internal passive controls are also often used in organizations. Unlike active controls, *passive* countermeasures are concerned not with directly stopping an attack, but instead with obtaining information about attacks so that future attacks can be stopped or the attackers identified. These types of controls are those referred to as detectives by Cavusoglu, Mishra, and Raghunathan (2004b). Passive countermeasures encompass the category of audit control and traffic monitoring from Ransbotham and Mitra (2009).

Examples of this type of control include intrusion detection systems (Mukherjee, Heberlein, & Levitt, 1994), audit logs (Schneier & Kelsey, 1999), and forensics (Corey, Peterman, Shearin, Greenberg, & Bokkelen, 2002). While passive countermeasures do not prevent attacks, they can be used as deterrents if their presence is known, and information gathered after an attack can be used to prevent further attacks through active means (Hansen, Lowry, Meservy, & McDonald, 2007). According to Richardson (2011), these countermeasures are used less often than active controls, and he hypothesizes that organizations are increasingly less interested in identifying the perpetrators of attacks and more in favor of stopping attacks outright.

The last category of IS security countermeasures available to organizations are external passive controls. As described by Tejay and Zadig (2012), these controls consist of the options available to attacked organizations within the criminal justice and legal system, to include referrals to law enforcement, the prosecution of attackers, and the various domestic and international laws pertaining to computer crime (Bell, 2002; Hollinger & Lanza-Kaduce, 1988). Unlike in the case of insider attacks, organizations rarely have access to external hackers for the purposes of administering punishment as they are usually out of reach (Foltz, 2004) and therefore must turn to the courts if sanctions are to be given. According to Richardson (2011), organizations are reporting fewer and fewer intrusions to the authorities, often because they did not believe that law enforcement would be able to help, because the incident was too small to report, or because they feared that negative publicity would hurt the stock price or public image of their organization. Other studies provide some support for the reduction in stock value following publicity of security breaches (Campbell, Gordon, Loeb, & Zhou, 2003;

Cavusoglu, Mishra, & Raghunathan, 2004a), as will be discussed next. Nevertheless, some research has shown that hackers fear arrest and prosecution and may change their behavior to avoid it (Hoath & Mulhall, 1998; Tejay & Zadig, 2012).

### 2.5 Economics of IS Security Breaches

As described above, organizations that have inadequate IS security countermeasures in place risk significant financial effects when breaches occur. A significant body of research has attempted to describe these effects, while other research has attempted to document the costs that arise when erecting countermeasures to prevent breaches from occurring.

Costs from breaches can take many forms. Kannan, Rees, and Sridhar (2007) describe two types of costs – *direct* costs and *indirect* costs. Examples of direct costs include losses of productivity by employees in the aftermath of an attack, the costs of notifying consumers who may have been affected by a breach, and the costs of communication with the media. Indirect costs, on the other hand, might consist of higher insurance premiums to affected firms or loss of future business due to a lack of trust in the attacked firm by consumers (Kannan, Rees, & Sridhar, 2007). Some attacks may have greater indirect costs than direct costs – for example, a defacement of a firm's website may have more indirect costs than the direct costs associated with an employee password breach of an internal company server due to the perception by customers of the insecurity of a more visible resource – the website. Such an attack may require significant and costly communications with customers and the press even though from a technical perspective the attack was less severe as no information was breached (Hancock, 2002). An extreme example of a financial effect of a defacement was the breach of the security company RSA's website in the early 2000s, which resulted in a 17% reduction in stock price, likely due to RSA's prominence in the information security field (Garg, Curtis, & Halper, 2003).

Hovav & D'Arcy (2004) examined computer virus attacks upon organizations and found that while 45% of affected companies suffered a negative impact upon share prices following the disclosure of the attacks, the majority of the companies did not experience a negative impact. They speculated that the market may anticipate virus attacks, or that reported damages from virus infestations – often in the millions of dollars, or billions industry-wide – are overinflated and the market effects are a reflection of their true impact. Garg, Curtis, and Halper (2003) offer another suggestion – that viruses, unlike other malware attacks, are generally untargeted and affect entire industries while not targeting specific firms. Massive virus infestations, while expensive to remediate, do not also typically involve the theft of confidential information and as such provide little for the stock market to react to (Garg, Curtis, & Halper, 2003).

Other types of attacks, however, have greater financial impacts for affected firms. Denial of Service (DoS) attacks, which affect the availability of IS from a "CIA" standpoint, have been shown to have noticeable same-day affects on share price (Garg, Curtis, & Halper, 2003). These DoS attacks often have significant upfront direct costs in terms of lost productivity and loss of commerce opportunities, and the share price of attacked firms that rely upon the Internet as the core of business are also more severely affected (Hovav & D'Arcy, 2003).

Breaches of internal company or consumer data, such as customer credit cards, have the most significant financial consequences (Campbell, Gordon, Loeb, & Zhou, 2003), with an average loss of between $17-28 million in market capitalization (Garg, Curtis, & Halper, 2003). One well-known example of a costly breach is the case of TJX Companies, in which hackers breached the network security of the firm in 2006 and stole over 45 million consumer credit cards from the company's corporate database. Direct and indirect costs from the breach ultimately reached $256 million, numerous executives and board members resigned, and earnings estimates following the attacks were down 57% (Xu, Grant, Nguyen, & Dai, 2008). Cavusoglu, Misha, and Raghunathan (2004a) found that costs of breaches are often even higher, with an average two-day market capitalization loss of 2%, or $1.65 billion, a trend that is confirmed in other studies (Goel & Shawky, 2009). Other research shows that the long-term effects from breaches, however, are not statistically significant, despite short-term impacts following the initial disclosure of a breach (Acquisti, Friedman, & Telang, 2006; Kannan, Rees, & Sridhar, 2007). Nevertheless, even short-term impacts may have significant financial implications for individuals and organizations that are trying to profit from the stock market in the short run.

To attempt to defray some of the costs following breaches and to change the perspectives of investors, Gordon, Loeb, and Sohail (2010) suggest an alternative approach for firms: voluntary and proactive disclosures related to IS security investments and events. The researchers examined a wide range of disclosures related to IS security, encompassing both positive disclosures such as the deployment of new security technologies and negative disclosures such as breaches, and found that disclosures which go beyond what is required by regulation have an overall positive effect on market value, of an average of 6% for the examined firms.

Gordon, Loeb, and Lucyshyn (2003) also suggested another method to “get ahead” of the costs associated with breaches – the proactive sharing of information regarding past breaches between firms. Such information sharing allows firms within the sharing alliance to collectively spend less on security while gaining knowledge from other members, which leads to increased levels of security. Gal-Or and Ghose (2005) also examined formalized information sharing alliances, such as industry ISACs and CERTs, and noted that even competing firms can achieve overall reductions in security costs, resulting in financial incentives to share data about attacks. Such an approach requires that monitoring occur so that members of the alliance are not seen as “free riders” that benefit from shared information without sharing themselves (Gordon, Loeb, & Lucyshyn, 2003). Moore, Clayton, & Anderson (2009) argue that similar information sharing needs to occur amongst security vendors themselves – while it does occur in the antivirus sector, such cooperation does not extend to other areas of IS security, such as phishing sites, counterfeit pharmacy sites, and other online fraudulent websites.

In general, security breaches have been shown to be common events (Gordon & Loeb, 2006) even though the rate of breaches has been difficult to forecast (Schechter, 2005); the question remains, how to allocate resources in defense? Gordon & Loeb (2002) suggested an economic model at odds with much of the IS security industry, stating that it made sense only to protect the midrange of security vulnerabilities, ignoring the very low or very high risk areas. They recommended that firms should only spend a fraction of the expected loss on security defense – never more than 37% of what the expected loss would be, but usually much lower than that. Due to the extraordinary costs of protecting everything at a high level of security, they suggest employing a cost-benefit approach and defending IS assets at a lower level of assurance. Willemson (2006) found that investments of up to 50% of the expected loss could be warranted, however. One problem with the Gordon and Loeb (2002) approach, however, is that many organizations are not quantifying their potential losses from security breaches, and Mercuri (2003) describes a number of quantitative approaches that organizations can utilize to estimate these losses.

In a similar vein, Cavusoglu, Mishra, and Raghunathan (2004b) developed a model to evaluate security investment decisions. This model evaluated the ideal placement of preventive controls versus detective controls and the optimal configuration of security technologies, including the design and pricing of IS security systems. They concluded that no single technology offers complete security and as such multiple technologies are needed, in a layered configuration often referred to as "defense in depth". Cavusoglu, Mishra, and Raghunathan (2004b) argued that hackers have been thinking and acting strategically about their attacks, and as such organizations should think and act strategically in the development of defensive countermeasures. They also suggested a game theory approach for security investment decisions, which results in the selection of security technology that affects the maximum cost savings for the organization.

In recognition of the inevitability of IS security breaches, Gordon, Loeb, and Sohail (2003) suggest another approach to reduce the cost of attacks – the employment of cyber-risk insurance. Such an approach requires that insurance companies attempt to address the problems of adverse selection and moral hazard in an IS security context. These policies tend to cover direct and indirect costs associated with attacks, and often include economic incentives for covered companies to increase their security. Cyber-risk insurance has the potential to alter the economic landscape of IS security defense because the investment in the insurance may mean that companies are comfortable with a higher level of risk from breaches (Gordon, Loeb, & Sohail, 2003).

There are other costs to firms from the risk of security breaches. Anderson et al. (2012) describe the billions of dollars of costs expended every year by organizations to defend against computer crime, including spending on countermeasures, direct and indirect losses of breaches, fines from governmental regulatory agencies following breach disclosures, and reduced market capitalization and loss of future business opportunity. All of these costs, according to Anderson et al. (2012), fund the continued growth of an underground economy of hackers seeking increasing financial returns from computer crime.

### 2.6 Summary

This review of IS security literature of relevance to attacks by external hackers has in essence described the attack cycle from start to finish, beginning with an examination of hackers and ending with the financial impacts of attacks. The literature described a steady evolution of hackers away from social motivations and toward financial motivations for attacks (Galbreth & Shor, 2010), as well as the increasing professionalism of hackers and hacking groups (Moore, Clayton, and Anderson, 2009; Ollman, 2008). While the dominant themes of research efforts have historically been focused upon deterring internal attacks (Straub, 1990; Willison & Backhouse, 2006) and upon whitehat defenders (Mahmood, Siponen, Straub, Rao, & Raghu, 2010), the description of the increasingly destructive and powerful abilities of external hacker attacks and hacker tools such as malware highlights the need to shift paradigms. A detailed analysis of the typology of IS security countermeasures (Ransbotham & Mitra, 2009) also highlighted areas where attackers may seek to innovate and discover new ways of executing attacks.

An analysis of the literature related to the economics of IS security attacks and breaches illuminated the devastating effects of hacker attacks upon organizations. Kannan, Rees, and Sridhar (2007) described direct and indirect costs that could be incurred from attacks. Garg, Curtis, & Halper (2003) examined the extensive financial effects of DDoS attacks and internal data breaches, which include damage to market capitalization and consumer confidence. Firms may also have economic incentives to examine alternative approaches, to include proactive security disclosures (Gordon, Loeb, and Sohail, 2010), security information sharing between firms (Gordon, Loeb, and Lucyshyn, 2003), and investing in cyber-risk insurance (Gordon, Loeb, and Sohail, 2003). Strategic approaches of organization security investments were also examined, with Gordon and Loeb (2002) recommending a cost-benefit approach toward deployment of security defenses around assets of varying risk and Cavusoglu, Mishra, and Raghunathan (2004b) analyzing a “defense in depth” approach to layering security countermeasures. The financial aspects of these attacks only underscore the need for an alternative approach to understanding how hackers develop their attacks.

# Chapter 3

# Research Design

## 3.1 Introduction

As described earlier, the issue of computer crime has been examined through a number of different lenses. These previous efforts include deterrence-based approaches (D'Arcy, Hovav, & Galletta, 2009; Straub & Welke, 1998), social approaches (Lee & Lee, 2002; Willison & Warkentin, 2013), and technical approaches (Ransbotham & Mitra, 2009; Cavusoglu, Mishra, & Raghunathan, 2004b). Many of these prior attempts focused upon individual computer criminals and neglected the interaction and evolution of the actors within a community (Lu, Polgar, Luo, & Cao, 2010). As such, the extant research has been insufficient to explain the continued success of the hackers and to understand the role that innovation within hacker communities plays in the defeat of IS security countermeasures.

This chapter will first introduce a theoretical framework which will form the basis of the present inquiry into innovation within hacker communities, followed by the derivation of a research model. The next section will describe the two-phased research method that was employed to gather empirical data, as well as the data analysis approach.

## 3.2 Theoretical Framework

### *3.2.1 Diffusion of Innovations Theory*

This research study adopts Diffusion of Innovations (DOI) theory as the theoretical basis to conduct this research. DOI theory has been employed by IS scholars to explain a wide variety of individual and organizational innovation adoption decisions (Baskerville

& Pries-Heke, 2001; Fichman & Kemerer, 1999; Mustonen-Ollila & Lyytinen, 2003; Straub, 1994), and formed the basis of this study. As described above, the argument for this research stated that hackers in communities innovate and disperse those innovations in a process similar to one employed by legitimate firms, and as such an organizational theory such as DOI is appropriate. This theory, first advanced by Rogers (2003), examines the process where an innovation is communicated through certain channels over time, among members of a social system. Rogers states that an innovation is "an idea, practice, or object that is perceived as new by an individual or other unit of adoption" (Rogers, 2003), and as such would encompass new hacking techniques that are being considered by a community of hackers.

Because of the focus upon the process by which an innovation is eventually adopted or rejected, innovation research can be thought of as a branch of technology acceptance literature (Karahanna, Straub, & Chervany, 1999). DOI in particular, with its emphasis upon the communication of innovations through a social system, is well suited to the study of the diffusion of IS security countermeasure-defeating technologies among hacking communities. In fact, Rogers (2003) has identified innovations within criminal circles as a potential application of his theory, and specifically mentioned computer viruses as an example of an innovation that has diffused very well through use of the Internet.

Rogers (2003) states that innovations perceived by members of a social system as having more of the following characteristics will be adopted more rapidly: the *relative advantage* of using the innovation over other technologies; the *compatibility* of the innovation with existing values, past experiences, and needs of the potential adopters; the *complexity* of the innovation; the *trialability* of the innovation, or whether it can be experimented with before complete adoption; and the *observability* of the results of the adoption to other members of the social system. Moore and Benbasat (1991) suggested the addition of two more characteristics: *image*, or the degree that the innovation enhances the user's social status within the social system, a characteristic they argue is separate from relative advantage; and *voluntariness*, the degree that an innovation's use is mandated or voluntary. Agarwal and Prasad (1997) found that of these seven characteristics, the complexity of the innovation is often not a significant factor, as some innovations are important enough that users will expend the necessary effort to overcome any usability issues.

By examining potential innovation adopters in light of the above characteristics, adopters can be classified on a scale of innovativeness, ordered by time to adopt the innovation: innovators, early adopters, early majority, late majority, and laggards (Rogers, 2003). The innovativeness of various adopter communities is an active strain of diffusion research within the IS realm (Cheng, Kao, & Lin, 2004; Fichman, 2001), as is comparing the adoption rates of the same type of innovation across different communities (Mustonen-Ollila & Lyytinen, 2003). The rate of adoption of innovations has been studied at length, and has generally been found to resemble an "S", with the innovators and early adopters at the start of the curve, the majorities in the middle as more and more individuals adopt the innovation, and the laggards at the end of the curve (Rogers, 2003). Rogers also describes the factors that influence the rate of adoption, to include: the five initial characteristics of an innovation described above, the type of innovation decision, the nature of both the communication channels and the social system, the extent to which change agents within the social system are promoting the innovation, and whether or not the innovation is being adopted by an individual or an organization, as organizations are typically slower to adopt.

Innovation diffusion is a process, one that ends with the decision to adopt or reject an innovation. Rogers (2003) describes the "innovation-decision" process as starting with knowledge, where an individual learns of the innovation's existence. Next, in the persuasion phase, the potential adopter forms a favorable or unfavorable attitude towards the innovation, followed by the decision phase, where the individual decides to adopt or reject the innovation. Next the innovation is deployed and used in the implementation phase, and finally the adopter enters the confirmation phase, where the individual seeks reinforcement of the decision to adopt the innovation, and may ultimately reverse the decision to adopt if negative feedback is received (Rogers, 2003). In these last two phases, the DOI approach differs from other theories in the adoption literature – notably the Technology Acceptance Model (TAM) and the Theory of Reasoned Action (TRA) – as DOI distinguishes between initial usage and sustained usage of an innovation, as opposed to simply determining usage intention (Agarwal & Prasad, 1997).

In addition to the diffusion of innovations amongst individuals, Rogers (2003) also relates that organizations can engage in this process. Particularly, Rogers states that virtual organizations are often more flexible and adopt innovations more rapidly. These virtual organizations – described as a "network of geographically-distant employees who are linked by electronic communication" (Rogers, 2003, pg. 405) – are analogues to the online hacker communities, which are made up of hackers that are often in different countries and who communicate using the Internet, that are the focus of the present study.

Organizational diffusion has been studied at great length by IS researchers (Baskerville & Pries-Heje, 2001; Fichman & Kemerer, 1999; Mustonen-Ollila & Lyytinen, 2003) and is a mature alternative to the study of diffusion between individuals. Rogers (2003) also describes the concept of organizational innovativeness and describes DOI research that has evaluated organizations based upon their level of innovativeness. As described in the review of the literature chapter, hacker groups and communities frequently operate like legitimate firms, including in areas of research and development and organizational structure, lending themselves well to a DOI-based organizational analysis.

The notion of hacker technique and tool innovation in particular is an appropriate area of focus for a study employing DOI. According to Barber (2001b), hacker techniques and tools are constantly evolving and becoming ever more complex. Bratus (2007) describes the emergence of new hacking techniques, and the development of specialized tools, in one type of hacker community – magazines such as Phrack. These techniques and tools allow hackers in the community to increase their flexibility and to constantly improve their attacks upon a victim's IS. The innovation of hacker techniques and tools, therefore, can be considered a technical process innovation, an area well studied by IS innovation scholars (Fichman & Kemerer, 1997; Mustonen-Ollila & Lyytinen, 2003; Nilakanta & Scamell, 1990).

### *3.2.2 Research Model*

The above discussion of the Diffusion of Innovations theory, as well as the examination of the two research questions guiding this study, have resulted in the derivation of the research model described in Figure 2.

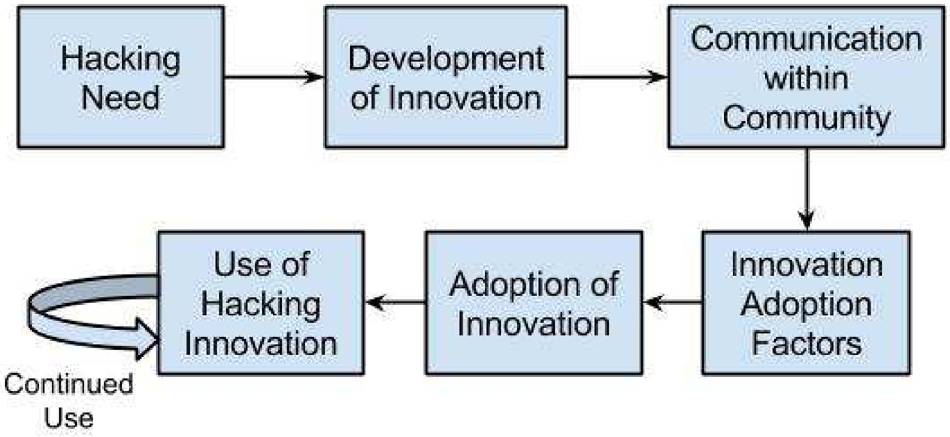

Hacking Need
Development of Innovation
Communication within Community
Innovation Adoption Factors
Adoption of Innovation
Use of Hacking Innovation
Continued Use

**Figure 2: Conceptual Model of Hacking Innovation Diffusion**

The components described in Figure 2 describe the process of the diffusion and adoption of hacking innovations. The first stage is the identification of a need for a new hacking tool or technique, perhaps because existing hacking tools or techniques are no longer effective against attacker attempts to defeat organizations' IS security countermeasures. The second stage involves the development of an innovation by member(s) of a hacking community, or the introduction of such an innovation developed outside the community into the community by a member. The third stage involves the communication of the hacking innovation by change agents within the community. Holt (2013) states that online hacker communities operate in a "relational J curve" structure, with a small number of users responsible for a large number of posts, which may be a useful method of identifying change agents. The fourth stage is the evaluation of the seven innovation adoption factors described above – relative advantage, compatibility, complexity, trialability, and observability as per Rogers (2003), as well image and voluntariness as per Moore and Benbasat (1991). The fifth stage is whether or not the innovation is actually adopted by the community, and the sixth stage involves the actual use of the new hacking innovation against victim organizations by members of the community. As the innovation is used, it is constantly evaluated by users to determine if it should continued to be employed, represented by the "continued use" arrow following the final stage.

Mustonen-Ollila and Lyytinen (2003) described two primary areas in the innovation process based on their analysis of DOI. The first area, *initiation*, corresponds to the first three stages of the conceptual model, located on the top of Figure 2. These first three stages also map to the first research question of this study. The second area described by Mustonen-Ollila and Lyytinen is *implementation*, which corresponds to the last three stages of the conceptual model, located on the bottom of Figure 2. These last three stages relate to the second research question of this study. Additionally, Rogers' (2003) innovation-decision process can be mapped onto the model in Figure 2. Rogers' "knowledge" phase encompasses the development and communication stages, the "persuasion" phase consists of the stage where the innovation factors are evaluated, the "decision" phase consists of the actual adoption stage, the "implementation" phase consists of the use stage, and Rogers' "confirmation" phase is reflected in the constant evaluation to continue using the innovation.

The use stage, as well as the process of evaluation or "confirmation" per Rogers (2003), is worthy of additional consideration. This final part of the diffusion of innovation process is described by Parthasarathy and Bhattacherjee (1998) as "post-adoption" behavior, and permits the identification of discontinuers and continued adopters, and should be considered as part of an evaluation of a successful innovation diffusion process. Parthasarathy and Bhattacherjee, in their analysis of adoption activity viewed through a DOI lens, determined that there may be an “expectation-reality” gap and found that later adopters of technology are more likely to discontinue, as early adopters rely more upon interpersonal influence when deciding to adopt an innovation. A related area to discontinuance is the body of research that seeks to understand IS project abandonment (Keil, 1995).

### *3.2.3 Philosophical Position*

This study, while utilizing a qualitative research method, resided squarely within the positivist epistemological paradigm. A qualitative research approach is appropriate for a positivist study, despite a common misconception that qualitative research is better suited for interpretive research (Klein & Myers, 1997) or, conversely, a misconception that positivistic research requires quantitative methods (Lee, 1989). In short, the positivist position assumes that reality is objective, and that empirically testable and falsifiable theories can be developed to reflect observations of this reality (Chua, 1986). Positivist ontology believes that there is an objective physical and social world that exists, whether or not individuals have any knowledge of it. Also objective is the researcher, who remains detached and neutral (Darke, Shanks, & Broadbent, 1998). Positivism is concerned with what individuals or organizations said or did, versus what the researcher thought they meant through the interpretation of symbols (Sarker & Lee, 2003).

Positivism is the dominant paradigm in IS research, and assumes a natural science ontology grounded in objective reality and frequently but not always involves the use of research techniques such as hypothesis and theory testing, quantitative variable measurement, and sampling of populations (Orlikowski & Baroudi, 1991). As such, this research should be evaluated by positivist criteria; evaluating one epistemological paradigm by the criteria of another is inappropriate, although such practice occurs frequently by reviewers and researchers unfamiliar with the paradigm being employed (Klein & Myers, 1997).

## 3.3 Research Methodology

This section details the data collection and data analysis methodology of the research. This research will consist of two phases – an expert panel to develop an initial list of hacking tool and technique innovations as well as an initial sample of hacking community sites, and a multi-site case study of online hacking communities to examine the diffusion of those tools and techniques identified by the expert panel. Such a two-step process involving an initial expert panel is commonly employed in management research designs (Jennings & Lumpkin, 1992).

### *3.3.1 Expert Panel*

As a preliminary phase to the research, a panel of recognized IS security experts were formed to help guide the next phase of data collection. The primary purpose of this panel was twofold: first, to develop a collection of key hacking technique and tool innovations of the past ten years, as well as associated keywords related to these innovations; and second, to identify influential hacking communities that will be appropriate sites for data collection in the second phase of this research. As this research sought to understand the process by which communities of hackers develop and diffuse new attack innovations, an initial list of the most disruptive innovations was an appropriate starting point for observing the hacking innovation diffusion process through all of the steps described in the research model above. The list of innovations provided by the expert panel was correlated with available security research literature to obtain technical details and other information about each innovation as necessary.

The panel of experts also provided an initial list of hacking forums that were evaluated for suitability as case study locations. As hacking communities are by their nature secretive, the research literature is not an appropriate place to derive a list of communities. Likewise, attempting to find hacking communities by Internet search engine may only turn up those that are not skilled at hiding themselves, and may not be representative of the communities of skilled hackers sought for this study. The experts that will comprise the panel had the expertise necessary to recommend a list of initial communities.

The expert panel consisted of eight IS security experts with specialties in computer crime investigation, hacker communities and culture, and IS security countermeasures. A panel with eight members was chosen following a review of IS studies that found that many of them employed expert panels of the same size (Aubert, Rivard, & Patry, 2004; Johnston & Warkentin, 2010; Wang, Liang, Zhong, Xue, & Xiao, 2012). Details regarding the background and experience of each panel member are described in a table in the next chapter. The formation of this panel also allowed for the clarification of any issues that arose regarding the innovations and hacking communities that were examined in the next phase (Sekaran & Bougie, 2010).

### *3.3.2 Case Study Approach*

The second phase of research consisted of a multi-site, qualitative case study. According to Yin (2008), a case study is "an empirical inquiry that investigates a contemporary phenomenon in depth and within real-life context" and "relies upon multiple sources of evidence, with data needing to converge in a triangulating fashion" (Yin, 2009, p. 18). Case studies take a holistic approach and are designed to examine a phenomenon from the viewpoint of the participants (Dubé & Paré, 2003; Tellis, 1997b). Unlike experimental methods, case studies do not manipulate independent variables or control confounding ones, but rely upon techniques such as questionnaires, coded interviews, or systematic observations (Boudreau, Gefan, & Straub, 2001). Case studies are not meant to be representative of a larger population, but are meant to understand the individuals or organizations within the cases themselves (Stake, 1995). It is important to note that case studies do not provide statistical significance, but instead link together many different types of evidence toward a relevant and strong conclusion (Runeson & Höst, 2009). Case studies within the IS discipline have a rich tradition, and allow the study of information systems in a natural setting so that the complexity of the process taking place can be better understood, shedding light on emerging topics (Benbasat, Goldstein, & Mead, 1987).

The procedures for case studies include the selection of cases, development of collection protocol, data collection and analysis, and the development of conclusions from collected evidence (Eisenhardt, 1989; Tellis, 1997b; Yin, 2009). Note that each of these procedural details will be discussed in subsequent sections. Additionally, case study methodology is an appropriate choice for research occurring within the positivistic paradigm (Darke, Shanks, & Broadbent, 1998; Dubé & Paré, 2003; Gibbert, Ruigrok, & Wicki, 2008; Lee, 1989; Sarker & Lee, 2002).

A single-case study is ideal when the researcher seeks to obtain a critical, extreme, or representative case to illustrate theory, while a multiple-case study – or what Stake (1995) terms a "collective" case study – permits literal and theoretical replication and is often considered to be a more robust methodology (Benbasat, Goldstein, & Mead, 1987; Yin, 2009). The present study undertakes a multiple-case approach across multiple hacker communities, and per Yin (2008) each case was analyzed separately, which allowed findings to be confirmed by replication logic. Following the separate analysis of each case, a final analysis of all the cases occurred. Such an approach allows both within-case and cross-case analysis (Dubé & Paré, 2003; Eisenhardt, 1989). Case studies involving multiple cases have been observed in prior DOI studies (Chircu & Kauffman, 2000; Mustonen-Ollila & Lyytinen, 2003; Reich & Benbasat, 1990). For the present study, four cases were explored; while previous innovation-related case studies such as Chircu and Kauffman (2000) and Mustonen-Ollila and Lyytinen (2003) only utilized three, Eisenhardt (1989) states that four is the minimum number required to achieve external validity through analytic generalization, which has been employed in other IS studies (Chetty & Holm, 2000). Four cases allow *literal replication* – that is, they permit the confirmation of similar results across cases. Important findings from the examination of one case can inform the analysis of later cases. Additional cases could be added during the study if divergent findings are observed and *theoretical replication* – that is, contrasting results but for reasons outlined within theory – is required (Yin, 2009);

however, this approach was not necessary for the present study. The unit of analysis for each case was each hacker community.

Data collection consisted of a triangulated approach to the study of the four initial hacker communities, and is described in greater detail in the proceeding section. Per Yin (2008), a data collection protocol was developed which will allow for a replicable and reliable instrument to be utilized across each case. This protocol included an overview of the case study project, the field procedures to be employed, the case study questions, and a guide for producing the final report. Details regarding the collection of data from interview subjects are provided below. The specific types of data encountered within each case include: postings made to the online hacker community under study; relevant news articles, court documents, or other secondary sources related to the hacker community; artifacts such as the hacker tools distributed within the community; and interviews of individuals who are active in the community. Reviewing postings made by hackers to such online communities is an “efficient and nonintrusive way to reach persons who attack computer systems” (Ransbotham & Mitra, 2009, p. 125) and allows the researcher to observe the interactions of the participants in their natural environment.

The triangulated data from each case were collected at each step in the conceptual model described in Figure 2. That is, data points regarding the diffusion of hacking innovations were reviewed from each data source – such as observations from web forum posts, interviews with hackers, analysis of court documents, and so forth – with an eye toward uncovering details of the innovation diffusion process. This process included the detailing of a need for a new hacking innovation, the development of the innovation, the communication of the innovation within the community, the factors influencing adoption of the innovation; the eventual adoption or lack thereof of the innovation, and the continued use of the innovation, as described in Figure 2.

The subsequent chart (see Figure 3), adapted from Yin (2009, p.59), will serve as a guide to the following detailed sections of the case study methodology, given that we have selected DOI as our starting point:

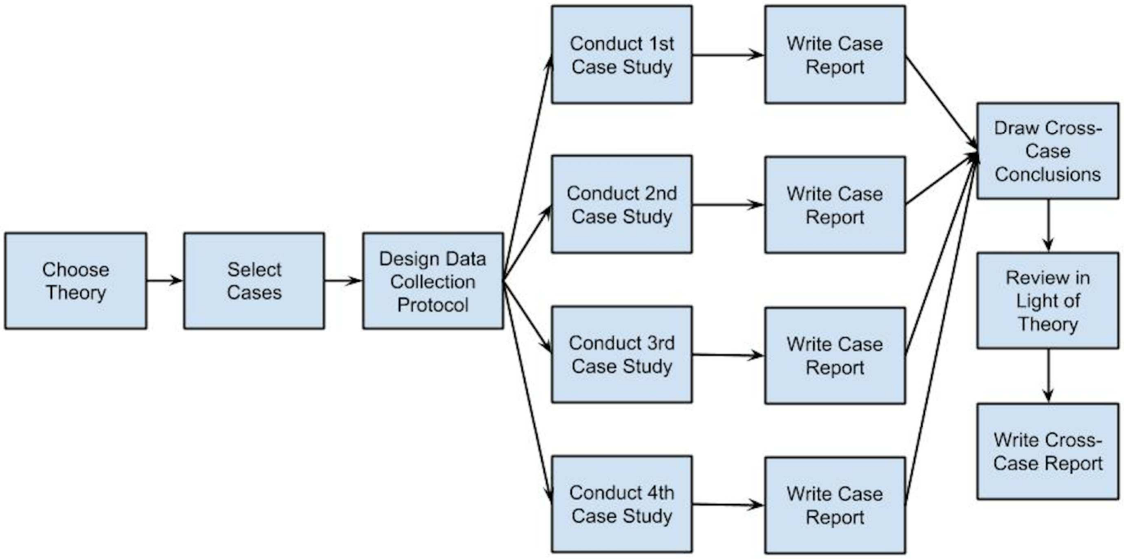


**Figure 3: Case Study Methodology (adapted from Yin, 2009)**

### *3.3.3 Selection of Cases*

As described above, the present study employed a multi-site holistic research design, with the sites consisting of four individual hacking communities, in the form of underground hacking web forums. The list of communities provided by the panel of experts was employed as a starting point; however, a preliminary examination was conducted of each community to determine if discussions of hacker tools and techniques are actually occurring, and one additional community was indeed sought. Yin (2009) recommends screening candidate case sites before engaging in data collection to ensure that they represent the desired behavior under study. This screening, according to Yin, involves querying knowledgeable people about the site – such as those in the expert panel – and perhaps obtaining quantitative data about the site to be sure it is suitable. For the present study, such quantitative data included the number of forum members, the number of total posts and threads, and the length of time the forum has been active. Forums with more activity, for a longer period of time, were more suitable than ones with less activity.

It should be noted that only hacker communities that are "blackhat" were examined, as the focus of this research is upon malicious hackers who are attempting to overcome the IS security countermeasures of victim organizations, which permitted the observation of participants in real-time. Data collection occurred within active, online hacker communities. These communities archived much of the older discussions that had occurred in prior years and allowed for the review of the early stages of an innovation's development. Such an approach also allows a longitudinal approach to the data collection, as samples can be taken from different points of time in the history of the hacking community. Due to researcher language restrictions, only English-language forums were reviewed.

Yin (2009) emphasizes that a multiple-case study should be analogous to an experiment repeated multiple times, where replication logic is emphasized. A multiple-case study should not be thought of as an analogy to multiple respondents in a survey or multiple subjects in an experiment, where sampling logic would dominate (Yin, 2009). Instead of employing a statistical approach to selecting respondents, this study sought to select a handful of illustrative cases to demonstrate literal replication of the theory, although attention will be paid for any instances of theoretical replication during cross-case analysis. Eisenhardt (1989) also states that cases should be chosen if they are theoretically useful, in that they replicate or extend theory by filling theoretical categories.

### *3.3.4 Case Study Protocol Design*

Now that the criterion for selecting the multiple sites has been described, the next step of the case study research design is to develop the *case study protocol*. The protocol "contains the instrument but also contains the procedures and general rules to be followed in using the protocol," and is "essential if you are doing a multiple-site case study" (Yin, 2009, p.79). The protocol provides a guide to the researcher for collecting data from each individual case. Having a prewritten protocol offers a number of benefits, including allowing the researcher to plan ahead for specific questions to be asked or observations to be made, providing a template for other researchers to review to ensure the study will collect the proper data, and it serves as a log where data collection and analysis can be recorded (Runeson & Höst, 2009).

According to Yin (2009), the following elements are included in the design of a case study protocol: an *overview* of the case study project, including the case study setting, the rationale for selecting the case, and the theoretical relevance; the *field procedures* to be used to collect data, including observations, interviews, and triangulation using historical or other sources, and a description of the tasks to include gaining access to the site, a schedule of collection activities, and providing for unanticipated events; the *case study* *questions*, which are the primary "instrument" used to collect data; and finally, a *guide for the case study report*, which provides an overview of how the data obtained from each case will be reported (Tellis, 1997b; Yin, 2009). The case study protocol employed for each case is produced in Figure 4, and will be explained below. It should also be noted that Stake (1995) recommends considering additional factors before gathering data, including the number of helpers, allocation of time, and possible expenses.

The case study overview is the first part of the protocol and contains introductory information regarding each case. This information includes details about each site setting – in the case of an online hacking forum, this would include the particular Internet address used to access the site and other details, such as the number of users, number of threads, and length of existence of each forum, as described in the previous section. Next, the rationale for selecting the case was provided, and answered the following question: what features make this site a good candidate for the case study? The theoretical implications of each site were also briefly described, and answered the following question: what will this site contribute to the case study?

The next part of the protocol describes the field procedures used in the study. First, access to the site must be obtained. In the case of an online hacking forum, the researcher may need to sign up for an account on the site if it is not publicly searchable. If the forum is located on a hidden network service such as Tor, as was the underground marketplace Silk Road, then logical access to the network will need to be gained as well (Biryukov, Pustogarov, & Weinmann, 2013). Once access is obtained, data collection can begin. Yin (2009) describes a number of sources from which case study evidence can be obtained, which are detailed in Table 2. This table also includes analogues to these sources applicable in an underground web forum environment.

- A. Overview of the Case
  - a. Case study site setting
    - i. Forum Internet address
    - ii. Statistical details of forum
  - b. Rationale for selecting case
  - c. Theoretical implications of the case
- B. Field Procedures
  - a. Access to particular site
  - b. Observation of forum participants (online / offline) in development of new hacking techniques and tools
    - i. Selection of innovations to be observed
  - c. Interviews of forum participants involved in the innovations
    - i. Description of interviewed individuals
    - ii. How individual was selected for interview – role in innovations, if any
  - d. Acquisition of data from other sources
    - i. News articles about use of hacking innovation, hacking websites, court documents, etc.
- C. Case Study Questions
  - a. Determination of hacking need
    - i. How is the problem defined?
    - ii. Does the need come first, or awareness of the innovation?
  - b. Development of innovation
    - i. Who engages in the process of developing the innovation?
    - ii. To what extent is this development collaborative or performed singularly?
    - iii. What are the roles of the various participants in the development of the innovation?
  - c. Communication of innovation
    - i. What are the communication sources for the innovation?
    - ii. By which channels do the communications occur?
  - d. Innovation characteristics
    - i. What is the relative advantage granted by the hacking innovation?
    - ii. How compatible is the innovation with the values, experiences, and needs of the community members?
    - iii. How complex is the innovation, and does increasing complexity dissuade potential adopters?
    - iv. How trialable is the innovation?
    - v. If the innovation is adopted, is it observable to other members?
    - vi. Does the innovation enhance the image of the user?
    - vii. To what degree is the use of the innovation voluntary?
  - e. Innovation Adoption
    - i. Is the innovation adopted or rejected by the community?
    - ii. What behavior and factors determine if an innovation will be adopted or rejected?
  - f. Implementation of Innovation
    - i. To what extent does reinvention of the hacking innovation occur, and for what reason?
    - ii. If the innovation is discontinued, why do the adopters discard it?
- D. Guide for Case Study Report

**Figure 4: Case Study Protocol**

**Table 2: Sources of Case Study Evidence**

| Source of Evidence | Examples from Yin (2009) | Web Forum Analogues |
|---|---|---|
| Documentation | Letters, email correspondence, announcements, news articles | Forum rules, policies, or terms of service; administrator announcements; news articles about forum or members |
| Archival records | Public statistical data, organizational records, maps/charts, survey data | Forum member lists, user profiles |
| Interviews | In-depth interviews, focused interviews, structured interviews | Interviews of participants, key informants |
| Direct observations | Real-time observation of events | Review of forum conversations (asynchronous) |
| Participant-observation | Assume roles within case study situation | [Not applicable for current study] |
| Physical artifacts | Tool, instrument, technology | Hacking tools derived from innovation |

Yin (2009) states that documentation is relevant to most case studies, mostly for the corroboration of evidence from other sources. In the present study, examples of documentation in a web forum environment consisted of forum rules or policies posted by administrators, as well as posted website terms of service. Forum administrators may also post administrative announcements in various forum sections. Furthermore, news articles or other Internet writings such as blogs may describe the activities of the forum or its members, and may be useful to corroborate other evidence found during the case study. Archival records, according to Yin, are another important source of evidence. In the present study, instead of statistical data such as census figures, an example of archival data was a roster of forum members or individual member profiles.

Yin (2009) also describes interviews as an essential source of case study data. In the present study, interviewees were selected from forum participants. The case study literature is generally silent on how many interviews to conduct in a study, and the number of interview participants in case study research can vary wildly (Mason, 2010); however, the goal is to obtain theoretical saturation (Runeson & Höst, 2009). For the present study, a total of two interview participants per community were sought. Interview participants were selected in a purposive manner and were contacted directly by the researcher based on forum posts regarding the innovations under study. The researcher approached community members who were senior members, who had been active within the community for a significant period of time, and who had experience with the innovations under study. Initial contact of interviewers identified through forum postings were made through the private messaging systems built into the community. In the initial contact, the researcher identified himself as an IS security researcher and requested an interview about hacking tools and techniques, and explained that the identity of the interviewee would be kept confidential. If the interview subject expressed interest, the researcher then provided the subject with a document describing the confidentiality process and rights of the interviewee, as shown in Appendix B. Note that the document described in Appendix B provides a brief overview of the purpose of the study, although care was taken with this overview to avoid introducing bias into the participant's responses. If after reviewing the document the interviewee decided to participate, the interview commenced, either over the private messaging system of the community, via an external chat service, or over email.

Yin (2009) states that there are a number of different types of interviews that can occur: the *in-depth interview*, where the interviewee provides opinions and insights, may act as a key informant who can provide access to other sources of evidence, and who agrees to spend time engaging in an interview for an extended period of time or for multiple interviews; the *focused interview*, where a person is interviewed for a short period of time, such as an hour, and where the questions are typically derived from the case study protocol; and a more structured interview, such as a *survey*, where quantitative data is produced. For the present study, a mix of in-depth and focused interviews was sought. An example of this interview technique can be found in Keil (1995), which employed a semi-structured interview where a detailed protocol was developed as part of the case study data collection. Runeson and Höst (2009) describe three strategies employed in case study interviews: the “funnel” model, which starts with open questions and moves to specific ones; the “pyramid” model, which begins with specific questions and ends with general ones; and the “time-glass” model, which begins with open-ended questions, moves to specific questions, and concludes with open-ended questions. One major barrier to interviews was subject reluctance; given that the participants of these communities are hackers and are possible criminals, some were hesitant to describe their activities to an outside researcher. This reluctance may be overcome in part by introductions from case study participants or members of the expert panel, as well as a description of the confidentiality process of the study (Stake, 1995; Yin, 2009).

The next type of case study evidence is direct observation. Yin (2009) describes this type as a contemporaneous observation of relevant behaviors or environmental conditions. Given that the cases for the present study were primarily historical in nature, as forum postings and activities are by their nature asynchronous and many of the innovations identified for study had taken place in the past, contemporaneous observation may not be feasible. However, for the purposes of this study, observation of innovation events within the hacker communities occurred as if they were transpiring in near real-time, so that the development of an innovation can be treated as a single event. Particular attention should be paid to the details, actions, and subtleties of the environment

(Benbasat et al., 1987). Yin also describes the participant-observation as another source of case data. In this type of collection, the researcher participates in the events under study. Such an ethnographic approach in a study of criminal hacker communities would be problematic from a legal and ethical standpoint and as such was not considered for the present study.

The final type of data source is the physical artifact. Yin (2009) states that in some case studies, an artifact – such as a tool, instrument, technology, or work of art – may be collected or observed. Yin relates that artifacts typically have less potential relevance in usual case studies. For the present study, however, the potential for the creation of an artifact is real, as a possible output of the hacker community innovation process is a new tool or exploit that can be used against victim organizations. As such, the collection of such an artifact – such as a hacker tool or source code – occurred if it was observed.

Yin (2009) describes a number of principles that should guide data collection in case studies. Multiple data sources should be utilized to allow for data *triangulation*. Triangulation occurs when different sources of evidence – such as interviews, archival data, and observations, for instance – converge upon the same facts. This permits the conclusions of the case study to be supported by multiple types of evidence, and addresses construct validity by demonstrating “multiple measures of the same phenomenon” (Yin, 2009, p.117). Triangulation is especially useful when collecting qualitative data, which can be broader or richer but not as precise as quantitative data (Runeson & Höst, 2009). Yin states that the concept of data triangulation can be troublesome for the researcher as techniques for collecting each type of data must be mastered, as opposed to other types of studies which may employ a single data collection method.

Yin (2009) also states that another principle of data collection is the use of a case study database. Unlike the report generated by the investigator at the conclusion of the case, the database records data collected during the study and can be the subject of secondary analysis, independent of the availability of the initial data. The database, according to Yin, should contain four primary components: the *notes* of the investigator, stored and organized in an efficient manner that can be understood by an outside party; the case study *documents* that may have been obtained during the data collection; *tabular materials* such as quantitative data that may have been collected; and finally, any *narratives* produced by the investigator, which Yin recommends be written as open-ended answers to the case study questions – which are described in the subsequent paragraphs – but not included in the final report. These narratives help the researcher make connections between and triangulate the different data sources for each case. The database also increases the repeatability and transparency of the research (Rowley, 2002).

The final principle of data collection, according to Yin (2009), is the development and maintenance of a *chain of evidence*. This process is similar to that employed by law enforcement and forensic specialists when collecting evidence for court cases and increases the reliability of the study. Through careful documentation, developing a chain of evidence would permit an outside observer to, using the same evidence, come to the same conclusions as described in the final case study report. This chain starts with the case study questions, then to the case study protocol, next to citations of specific evidence in the case study database, and finally to the case study report itself (Yin, 2009). A reader should be able to start at either end – the initial questions or the final report – and following the chain of evidence in either direction, reach the other end (Dubé & Paré, 2003).

The next part of the protocol describes the actual questions of the case study. Yin (2009) states that this is what is commonly referred to as the “instrument”. However, these questions are not asked to interviewees or other case study participants; instead, they are directed toward the investigator, and serve as a structure of inquiry to remind the researcher of the specific data that is to be gathered during each case. Each question should also detail the sources of evidence where data is likely to be obtained. The questions should also be directed to the unit of analysis of the case study – that is, the hacking community – and not the unit of data collection – that is, the individual hacker participants (Yin, 2009). As described in Figure 4, the following case study questions were considered as part of the protocol. A detailed description of each question, and the linkage back to the propositions and research model are described below. The case study questions are organized into sections, each section corresponding to a concept expressed in the research model in Figure 3.

The sequence of questions below were applied for each innovation studied within each case. The initial round of innovations that will be analyzed using these questions will be obtained through the expert panel, although additional innovations were added if needed as they were encountered during data collection within each case. Refer to the questions in Figure 4, the rationale of which is described in detail below.

The first set of questions deal with the determination of a *hacking need* by members of the community, for new hacking tools or techniques. Rogers (2003) defines a need as a "state of dissatisfaction or frustration that occurs when an individual's desires outweigh the individuals actualities" (Rogers, 2003, p.172), and identifies that need recognition may occur when future problems are anticipated, or if a social system prioritizes a problem high on an agenda. This is part of the first stage of Rogers' innovation-decision process, the "knowledge" state, as described earlier. Of particular interest is how the systems under study – the hacking communities – define the need for an innovation. Much of the extant DOI research does not describe this aspect of innovation, and instead selects a particular innovation to be studied – such as ERP system development (Bradford & Florin, 2003), travel reservation system adoption (Chircu & Kauffman, 2000), or Microsoft Windows deployment (Karahanna, Straub, & Chervany, 1999) – and does not describe how and why the organization decided that an improvement was needed. The notion detailed in proposition P1, that an innovation is developed when existing hacking methods no longer succeed in breaching targeted organizations, will serve as the underlying assumption for this part of the innovation process. Interestingly, Rogers (2003) states that a need can prompt the development of an innovation, or vice versa, an innovation may prompt a need in an individual. For each innovation reviewed in each case, identifying the origin of the need determination will prove illuminative – was the innovation developed as a result of a hacking need within the community, or did the community members learn of an innovation and subsequently develop a need for it?

The second set of questions relate to the actual *development* of the innovation under study. Rogers (2003) defines development as "the process of putting a new idea in a form that is expected to meet the needs of an audience of potential adopters" (Rogers, 2003, p.146). If the innovation is developed within the organization, who participated in the development? Was the innovation developed externally to the community and discussed or debated within the community, or was it developed wholly within the hacking community? Previous DOI research has studied both in-house development efforts (Mustonen-Ollila & Lyytinen, 2003) and innovations that were developed externally to the organization (Karahanna, Straub, & Chervany, 1999). If the development is conducted within the hacking community, is it a collaborative effort or does a single individual develop the innovation? If collaboratively, what are the various roles that the participants play in the development? Understanding how the innovation was developed is a necessary component of the overall understanding of the eventual diffusion and adoption of the innovation.

The third set of questions deal with the *communication* of the innovation within the hacking community. Communication is also part of the "knowledge" stage of Rogers' (2003) innovation-decision process. Rogers states that participants acquire knowledge of an innovation through a "communication channel" consisting of a source, such as an individual or organization who originates the communication, and a channel, consisting of a means from which the message travels from the source to the recipient. Communications channels can be *interpersonal*, that is directly from one source individual to one recipient, versus *mass media*, where a source or small number of sources can broadcast to a large audience or recipients. Rogers generalizes that mass media channels are more important at the knowledge stage of the innovation-decision process, whereas interpersonal channels are more important at the persuasion stage. Similarly, these channels can also be *cosmopolite*, where communication sources originate outside of the social system, versus *localite*, where the sources originate within the social system. Again, Rogers generalizes that cosmopolite channels are more important at the knowledge state, while localite channels are more important during the persuasion stage.

These concepts of the communication channels used for the innovation lead to the following questions: what are the communication sources for the innovation, and by which channels do the communications occur? Within the DOI literature, Zmud (1983) examined cosmopolite channels and their impacts upon software development teams, and found that these external sources were effective if the internal environment of the group is supportive of innovation. Rogers (2003) also found that the effectiveness of these channels varies depending upon the categorization of the adopter by time to adopt, from early adopters to laggards as described previously. Rogers states that in general, early adopters are more affected by mass media and cosmopolite channels, while adopters toward the laggard end of the scale are more influenced by interpersonal and localite channels.

The above questions relating to the establishment of a need and the development and communication of the innovation were all part of Rogers' (2003) knowledge phase of the innovation-decision process. The next set of questions relates to the persuasion phase of the process and deal with the characteristics of the innovation. The first question asks, what is the *relative advantage* granted by the hacking innovation? If the innovation outperforms its precursor, then users may be more likely to adopt it. The second question asks, how *compatible* is the innovation with the values, experiences, and needs of the community members? A more compatible innovation is less uncertain to the adopter and may fit their needs better (Rogers, 2003). The third question asks, how *complex* is the innovation, and does increasing complexity dissuade potential adopters? Recall that Agarwal and Prasad (1997) in their research found that complexity was not a significant factor; however, Thong (1999) found that relative advantage, compatibility, and complexity are all significant factors, indicating that it is worthy of further investigation. The next question asks: how *trialable* is the innovation? Whether or not the hacking innovation can be tested before it is adopted may have impact upon the eventual adoption of the innovation. Furthermore, if the innovation is adopted, is it *observable* to other members? Observability allows other members of the community to view the innovation at work and may further encourage wider adoption (Rogers, 2003). Moore and Benbasat's (1991) last two characteristics round out the questions for this section: Does the innovation enhance the *image* of the user? Finally, to what degree is the use of the innovation *voluntary*? This last question may have limited relevance in hacker communities, which may be considered to be looser social structures than an employee working in a firm; although to the extent that these communities do imitate legitimate firms, there may be some rules of behavior governing the use of the innovation.

The next set of questions relate to the decision phase of the innovation-decision process. At this stage, the innovation can either be adopted or rejected. According to Rogers (2003), rejection can itself take two forms: active rejection, where an innovation is first considered or perhaps tested before it is rejected; and passive rejection, where an innovation is rejected outright without being considered. This leads to a case study question: Is the innovation adopted or rejected by the community? This question leads to another case study question: What behavior and factors lead determine if an innovation will be adopted or rejected?

The final set of case study questions pertain to the actual use of the hacking innovation, and involves both the implementation and confirmation phases of Rogers' (2003) innovation-decision process. During the implementation phase, the innovation is actually employed by the adopter, and various issues may be overcome, such as how to obtain the innovation and how to use it properly. Eventually, the innovation becomes institutionalized and is no longer considered a "new" idea. The concept of *reinvention*, or the modification of an innovation by an adopter during its use, is introduced during this phase. Instead of passively using an innovation as-is, some adopters may change it to suit their needs better. Rogers states that innovations that are reinvented by users are more likely to be adopted at a faster rate and experience continued and sustained use. This leads to another case study question: To what extent does reinvention of the hacking innovation occur, and for what reason? During the final stage, confirmation, the adopter seeks reinforcement of their decision to adopt the innovation, and may decide to discontinue use of it if this reinforcement is not obtained (Rogers, 2003). Discontinuance may occur because an adopter becomes disenchanted with the innovation, and Rogers suggests that later adopters discontinue use at a higher rate than earlier adopters. This tendency of later adopters to discontinue was confirmed by Parthasarathy and Bhattacherjee (1998) during their examination of online service adoption and discontinuance. This leads to an additional case study question: If the innovation is discontinued, why do the adopters discard it?

Now that the case study questions have been defined for each case, Yin (2009) states that the final part of the case study protocol is the development of a guide for the case study report. Tellis (1997b) states that some researchers prefer to keep their report in a journal format, while others prefer a more structured system. Each individual case should have its own report. The present study will record the report in outline form with headings for each case study question. Runeson & Höst (2009) refer to this section of the protocol as the "data analysis guidelines" section. The analysis of the data will be discussed in the next section.

### *3.3.5 Data Analysis*

With the development of the case study protocol, including data collection and instrument, data analysis can now be discussed. Data analysis is intended to focus upon patterns within the collected case study data, so that conclusions can be drawn. However, data collection and analysis are not intended to be completely separate phases – analysis and data collection should overlap so the analysis can guide future data collection, and allow the researcher to remain open to new ideas or patterns that may be encountered in the data (Darke, Shanks, & Broadbent, 1998). Within the case study methodology, data analysis design consists of two major steps – selecting an appropriate analytic strategy, and then choosing an appropriate analytic technique. Whichever strategy and technique is selected, the overall goal is to become intimately familiar with each case, so that unique patterns of each can emerge and be examined (Eisenhardt, 1989).

There are four analytic strategies for case studies, according to Yin (2009), which can be used singularly or combined. These include: relying upon theoretical propositions, which are derived from research questions and a thorough study of the literature; developing a descriptive framework, which is primarily used if data is collected before propositions are developed; employing both qualitative and quantitative data, and subjecting the quantitative data to a robust statistical analysis; and examining rival explanations to determine if they can explain the observed behavior, which requires that the rival explanations be identified ahead of time so data supporting them can be collected during the data collection phase. The present study will employ a theoretical propositions approach (Tellis, 1997a) combined with a thorough examination of rival explanations. A detailed statistical analysis is not applicable given the types of evidence that will be collected, as described in the previous section; additionally, a descriptive framework is not appropriate given that propositions have been considered prior to the initiation of data collection.

The concept of rival explanations deserves further explanation. Yin (2009) provides a number of reasons for why a rival explanation may provide the explanatory basis for observed behavior, and breaks these reasons into two types –"craft" rivals, and "real-life" rivals. A craft rival is an explanation caused by errors in research design, and Yin includes the following three: the null hypothesis, where the behavior is explained by chance circumstance only; threats to validity, which are detailed in the next section of this chapter; and investigator bias. Yin also describes six real-life rivals: a direct rival, where an intervention other than the expected intervention explains the results, which is described in the next paragraph; a commingled rival, where a combination of the expected intervention and other interventions together explain the results; an implementation rival, where the process of implementing the intervention explains the results, instead of the intervention itself; a rival theory, where a completely different theory better explains the results; a super rival, where some activity larger than but including the examined intervention explains the results; and finally, a societal rival, where social changes and trends explain the results, and not an intervention. The present study must be sure to collect enough data that these types of rival explanations can either be supported or refuted as necessary.

Each of the propositions described previously can be explained with direct rival propositions. The original propositions of this study are included below (P1-P4), along with sample direct rival propositions (R1-R4) that may also offer explanation. Note that other rival propositions are also possible. Again, data collected must be broad enough to support or discount these rival propositions.

*P1: Participants of hacking communities develop new techniques when existing hacking techniques cease to reliably breach targeted organizations.*

*R1: Participants of hacking communities develop new techniques as a means of demonstrating skill and earning status in the communities.*

*P2: The methods by which the new hacking techniques are communicated amongst the community affect the adoption of the innovations.*

*R2: A hacking innovation, if useful enough, will be adopted by the community regardless of the manner in which it is communicated through the social system.*

*P3: Change agents within the hacking community are instrumental in the adoption of the hacking innovation.*

*R3: The role of change agents within a community advocating for particular innovations is minimal; instead, the merit of the innovation dictates its adoption.*

*P4: Continued use of a hacking innovation by community members will determine the ultimate adoption of an innovation.*

*R4: Ultimate adoption of an innovation is unaffected by continued use of the hacking innovation.*

Analytic techniques are the next phase to consider when selecting a data analysis method. Yin (2009) describes a total of five techniques – pattern matching, explanation-building, time-series, logic models, and cross-case synthesis. The present study will employ a combination of pattern matching, explanation-building, and cross-case synthesis, and not logic models or time-series, as are described below.

One of the main analytic techniques for case studies is the concept of *pattern matching*. This technique compares a pattern of empirically-observed outcomes against a pattern that has been derived theoretically (Yin, 2009). This pattern may consist of "a situation where several pieces of information from the same case may be related to some theoretical proposition" (Tellis, 1997b). Such an approach requires that the pattern that is expected be defined before data is collected. Yin states that pattern matching is the most desirable of the case study analytic techniques, and as such it was employed in this study.

For a pattern matching approach, there are some considerations that must be understood before it is employed. Borrowing from natural science terminology, Yin (2009) states that one technique is using non-equivalent dependent variables as a pattern. For each outcome studied, if the values predicted in the proposition were found while alternative patterns of values – including those from validity threats – are not, this is a pattern that allows strong causal inferences to be made. This requires that threats to validity be identified ahead of time and that numerous patterns be analyzed to ensure that the validity threats so not account for the observed pattern. To better discern patterns, Stuart et al. (2002) suggest varying the order in which data from the cases is arrayed, grouping like and unlike events together, so commonalities and differences may become evident.

A related approach is to conduct tests on the rival propositions described above to determine if observed patterns are a result of those propositions, instead of the theoretically-derived ones. Yin (2009) also notes that because pattern matching is not subjected to statistical analysis, precision may be low. Yin suggests postulating less-subtle patterns to avoid interpretation errors on the part of the researcher in either confirming or rejecting a pattern match. Within the IS discipline, Sarker & Lee (2003) utilized a pattern matching approach in their case study analysis of ERP implementations. Keil (1995) also employed pattern matching to identify factors in his study of software project management.

Another analytic technique is the concept of *explanation-building*. This approach attempts to explain how or why a certain observed behavior occurred (Yin, 2009). Tellis (1997b) describes explanation-building as another form of pattern matching, where the analysis of the study is conducted by constructing an explanation of the case. This approach is more suited for explanatory case studies than exploratory ones, and allows the explanation to be built over the analysis of multiple cases, so that the final explanation derived at the conclusion of the study may not have been stipulated at the outset, unlike the pattern matching technique described previously. These explanations typically take narrative form and reflect propositions (Yin, 2009). This technique was also employed in this study.

Developing explanations is an iterative process: an initial statement or proposition is defined, then an initial case is examined, and based upon the results of comparing the initial case against the proposition, the proposition is revised. The proposition is then compared against a second, third, or more cases, and is revised after each; and is repeated until no additional insights are gained (Tellis, 1997b; Yin, 2009). Yin cautions that the explanation-building approach is more difficult and the repeated iterations may cause the researcher to drift away from the original topic of interest, although a safeguard against this tendency is to constantly refer to the original research goal. Other safeguards include reliance upon the previously-discussed tenets of a successful case study – the use of a case study protocol, a case study database, and maintaining a chain of evidence. Within the business research arena, Chetty and Holm (2000) employed both a traditional pattern matching and explanation-building approach in a multiple-case study of international network behavior among firms; explanations for observed behavior from each case were built iteratively, and resulted in a final explanation regarding internationalization.

The third type of analytic techniques is the *time-series analysis*. This approach directly mirrors the technique by the same name in experimental and quasi-experimental analysis (Tellis, 1997b), and involves tracing changes in variables over time. Time-series analyses can consist of simple series, where only one dependent or independent variable is observed and an observed trend is compared to a theoretically-anticipated trend, or to a rival trend. A complex series analysis may involve not just the rise or fall of a particular variable, but possibly the rise and then fall of a variable within a single case, or other mixed pattern, and may involve multiple variables (Yin, 2009). Chronologies can also be a form of time-series analysis, where the placement of specific events in a case in chronological order may lead to causal conclusions, if the sequence of events follows a predicted sequence instead of a rival sequence. Time-series analyses may be used in either single- or multiple-case studies. Tellis (1997b) points out that time-series analyses can be problematic as variables may change multiple times or for multiple reasons, which may make the starting or ending points of an event difficult to identify. Such a technique is not suited to the present study as the diffusion of innovations to be examined does not have an explicit time component, and a time-series approach often involves the introduction of an improvement or intervention (Runeson & Höst, 2009). An example of a time-series analysis within the IS discipline can be found in Keil's (1995) longitudinal study of IS project failure.

Another type of analysis is the employment of *logic models*. This approach describes a complex series of events over a period of time – but unlike the time-series analysis, logic models are described as a chain of cause-effect-cause-effect patterns (Yin, 2009). In a logic model approach, a dependent variable event in one stage becomes an independent variable causal event at the next. In essence, a logic model approach is another form of pattern matching, as the observed events are compared against a theoretically predicted sequence of events. Depending upon the unit of analysis, logic models can be applied at the individual, organizational, or program levels (Yin, 2009). A logic model approach is not ideal for the present study, however, because they are most appropriate for studies that involve the introduction of an intervention or significant change to the subject of the case study.

The final type of analysis is the *cross-case synthesis*, which is designed specifically for a multiple-case study. With this technique, each individual case is first treated as a separate study. If a large number of individual case studies are present, the findings from each can be analyzed using statistical analysis techniques, and conclusions drawn; if data from a smaller number of cases is collected, as anticipated in the present study, other techniques including the employment of word tables may be used (Yin, 2009). Word tables involve the placement of data from the individual cases into an array to look for patterns. Multiple word tables are created for the areas of interest to the study, and data from across the multiple cases can be compared. Yin states that this approach requires argumentative interpretation of results instead of simple numeric tallies, and describes cross-case synthesis to be analogous to cross-experiment interpretation, which – even in a natural science, experimental, and statistical setting – requires interpretation when only a small number of experiments have been conducted. This analytic technique was also employed in the present study.

Eisenhardt (1989) suggests developing the categories or dimensions to be compared both from literature as well as patterns observed in the case study data. These dimensions can be placed in tables – such as a 2x2 grid – to compare several categories at once, or pairs of cases can be compared against each other to identify similarities or differences of each case. Such forced comparisons could result in the development of new data categories or new concepts worthy of study (Eisenhardt, 1989). An example of cross-case analysis in IS can be found in Gable (1994), a study of enterprise IS adoption where five main variables were identified in each case, which corresponded with patterns predicted in literature. The values of these five variables were extracted from each case and compared against each other, and against patterns suggested in the literature, to obtain a deeper understanding of the issue as a whole.

To ensure that analysis is performed in a high-quality fashion, Yin (2009) provides four principles that should be followed. First, the investigator must show that the analysis relied upon all of the relevant evidence obtained during the study. Second, rival interpretations of the evidence should be included and addressed to demonstrate that they do not account for the observed behavior. Third, the analysis should be sure to address the most significant aspect of the case study – the underlying goal of the original research. And finally, the researcher's own prior expert knowledge of the subject should be utilized to enhance the analysis (Tellis, 1997b; Yin, 2009).

The actual procedure to analyze the collected data should also be discussed. There are multiple ways that case study data can be analyzed, including the employment of transcripts, tabular displays or graphs, or sequence analysis for time-series approaches (Eisenhardt, 1989). Darke, Shanks, and Broadbent (1998) suggest three areas: data reduction, where data is selected and simplified; data display, or organizing the data into narratives, tables, charts, or graphs; and conclusion drawing and verification, where meaning is drawn from the data, and a logical chain of evidence is constructed. To facilitate this analysis, this study will employ the coding of raw data for data reduction. Various types of data displays will also be employed to both demonstrate the chain of evidence and to assist in the uncovering of connections in the data, as will the usage of quotes from the participants of the communities under study, so that the voice of the participants can be heard (Dubé & Paré, 2003).

Runeson and Höst (2009) recommend analyzing raw qualitative data by developing a series of codes, where parts of the collected texts are assigned a code that represents a theme, area, or construct of relevance to the study. It is possible to have a hierarchy of codes and sub-codes to assist the analysis. Codes are an efficient way to conduct data reduction and to validate interpretations of data (Dubé & Paré, 2003). Codes can be applied to different units of data, including whole episodes, interviews, or documents (Stake, 1995). Following the assigning of codes to the collected data, the coded data is tabulated by organizing the codes into individual tables, much as the concept of word tables described by Yin (2009). Runeson and Höst (2009) describe four approaches to coding: immersion approaches, which are the least structured and rely upon the intuition and interpretive abilities of the researcher; editing approaches, where *a priori* codes are developed during analysis; template approaches, where codes are developed ahead of time based upon research questions; and quasi-statistical approaches, which eschew interpretation and rely upon the calculation of frequencies of particular words or phrases within texts. Runeson and Höst states that immersion and quasi-statistical approaches are quite difficult, and therefore most researchers employ editing and template approaches. Yin (2009) takes a template approach, stating that codes that are developed should have a logical progression from the study's research questions.

Bandara, Gable, and Rosemann (2005) used a combination of an editing and template approach in their case study of business process modeling. They first coded behavior that was related to theoretical constructs (template approach) and identified new constructs within the data (editing approach). Bandara et al. then refined the codes to determine if the text referred to the mere existence of a construct or was evidence of the criticality of a construct. Using qualitative analysis software, they identified keywords of relevance and built sub-constructs, and grouped codes into summary matrices for pattern matching and comparison.

Keil (1995) also employed coding of case study data as an analysis approach. First, a narrative of each case study was developed, as described previously. Then a table of key information was developed from the narrative and the relative information – in Keil's case, project management information available to decision-makers – was coded in terms of the availability of the information for each project. As another example of coding, Boudreau, Gefan, & Straub (2001) conducted coding of concepts in their meta-analysis of IS research articles, where the description of particular research methods or tools were noted and tabulated.

### *3.3.5 Threats to Validity*

The next section of the research methodology chapter is a discussion regarding various validity threats to the present study. In general, there are four threats to validity that must be considered and defended against: construct validity, internal validity, external validity, and reliability (Gibbert, Ruigrok, & Wicki, 2008; Runeson & Höst, 2009, Tellis 1997a). Each of these aspects will be considered below.

The first area of concern is *construct validity*, or how the data being studied reflects both the aims of the researcher and what is examined through the research questions (Runeson & Höst, 2009). This threat is prevalent during the data collection phase of the research (Gibbert, Ruigrok, & Wicki, 2008) and seeks to ensure that the operationalization of a concept actually measures what it is supposed to (Boudreau, Gefan, & Straub, 2001). To overcome construct validity issues, three main approaches can be employed: triangulation of multiple data sources, having key informants review the case study report, and maintaining a clear chain of evidence (Sarker & Lee, 2002; Yin, 2009). As described above, data triangulation was achieved by collecting data from multiple sources within the community, such as interviews, observations, documents, and so forth. A chain of evidence was constructed as described previously so that readers of the research will be able to follow the path of the study, starting with the collected data and finishing with the conclusions.

The next threat to validity is that of *internal validity*, which is a concern during the data analysis phase of research (Gibbert, Ruigrok, & Wicki, 2008) within explanatory studies (Tellis, 1997a). This threat arises when a causal relationship between two variables is claimed to be sure it is not caused by a third variable, or if the absence of such a relationship means that the event does not occur (Runeson & Höst, 2009; Sarker & Lee, 2002). Essentially, we need to show that variable X leads to variable Y, and that Y was not caused by variable Z (Gibbert, Ruigrok, & Wicki, 2008). To counter this threat, Yin (2009) recommends employing pattern matching logic as previously described – if empirically observed patterns match theoretically predicted ones, but not patterns from rival theories, then internal validity is high (Sarker & Lee, 2002). This study did indeed employ pattern matching logic to counter the threat of internal validity. Pattern matching should occur both for literal and theoretical replications (Stuart et al., 2002). A clear research framework can also help counter this threat (Gibbert, Ruigrok, & Wicki, 2008).

Another threat to be considered is that of *external validity*, or the extent to which the case study findings can be generalized beyond the immediate case (Stuart et al., 2002; Tellis 1997a). Sarker and Lee (2002) note that generalizability applies to the theory being tested and not to the individual case studies themselves. Furthermore, statistical generalization is not possible with case studies, but analytic generalization is achievable (Gibbert, Ruigrok, & Wicki, 2008; Stuart et al., 2002). To achieve analytic generalizability, Eisenhardt (1989) recommends a minimum of four cases so that robust cross-case analysis can be performed, which the present study employed. To further enhance generalizability, sufficient details should be provided for the case study selection and context so that the reader will comprehend the sampling choices employed (Gibbert, Ruigrok, & Wicki, 2008). This approach was employed in the present study.

The final threat to validity comes from *reliability*, or the extent to which the case study's procedures can be repeated for the same result (Stuart et al., 2002). There are two main aspects to this threat – transparency and replication. Transparency is assured through the development and description of a case study protocol, as described above. Replication is assured by maintaining a case study database for collected evidence, consisting of case study notes, documents, tabular material, and study narrative, all of which is available should additional researchers attempt to replicate the results (Gibbert, Ruigrok, & Wicki, 2008; Sarker & Lee, 2002; Stuart et al., 2002). Additional methods to improve reliability come from clear coding of collected data and clear interview questions (Runeson & Höst, 2009).

### 3.4 Summary

This chapter detailed the overall research design of the study. The DOI theory and applicability to IS security was described, and a research model of DOI applied to hacking communities was derived. The philosophical positions of the research were provided, and then an appropriate research methodology that reflected those positions and incorporated previous DOI and IS literature was selected. This methodology included an expert panel and a case study approach of four hacking communities. The case study protocol was described in detail, and finally, data analytic strategies appropriate for the study were described.

# Chapter 4

# Results

## 4.1 Introduction

This chapter describes the execution of the expert panel and the four-site case study undertaken for this research, as well as the results from each. Following the description of individual results of each case, a section describing cross-case analysis will be presented.

## 4.2 Data Analysis

### *4.2.1 Results of Expert Panel*

As described in the Methodology section, a panel of information security experts, with specific experience in computer crime, was convened prior to the initiation of data collection. The table below describes these panelists:

**Table 3: Expert Panel Participants**

| **Panelist Description** | **Panelist Location** |
|---|---|
| Law enforcement agent specializing in computer crime | Southwestern US |
| Information security consultant | Northwestern US |
| Researcher specializing in computer crime | Southeastern US |
| CEO of information security company | Northwestern US |
| Threat intelligence expert in financial services industry | Mid-Atlantic US |
| Threat intelligence consultant | Northeastern US |
| Antivirus industry expert | Canada |
| Antivirus industry expert | Ireland |

The panel engaged in a discussion regarding recent hacking tool and technique innovations that would be likely to be discussed in underground hacker communities and be suitable for detailed study in this research. The panel derived the following list of innovations:

**Table 4: Expert Panel Innovations**

| **Innovation** | **Additional Details from Panel** |
|---|---|
| Havij | SQL injection technique |
| “As a service” model of cybercrime | Examples: DDoS as a service, spam as a service. “The big advantage of that is that you do not need to understand how the service works - as long as you pay the money, and the guy has a good reputation - it just plugs effortlessly into your setup.” |
| Bulletproof hosting | Webhosting specifically for cybercrime |
| Cryptocurrency | Stealing bitcoins, mining bitcoins, or steganography in bitcoin wallets |
| Spyware | Evolution: Spyware (circa 2006/2007) -> Fake antivirus -> ScareWare -> Ransomware |
| Point-of-sale malware | For stealing credit card data from retailers |
| Cybercrime-enabling services | Counter anti-virus services, crypting as a service – “fully undetectable” |
| Memory resident tools | An anti-forensics technique |
| Shodan search engine | Embedded systems / SCADA hacking |
| Cross-platform or mobile malware | Consequence of shift away from Windows operating systems to Mac / mobile |
| DNS Reflection attacks | Type of DDoS attack |
| Booter services | DDoS method popular with gamers |
| Amplification attacks | DNS, SSDP, NTP, other types of amplification attacks for DDoS |
| Malware services | Customer-focused - support agreements, FAQs, customer forums |
| Affiliate programs | Methods for malware distribution |

The innovations described in Table 4 were employed in the case study portion of the data collection. Not every innovation was observed or popular within every hacking community studied; some communities had specific focuses, or others had members who were familiar with certain techniques but not others. However, the above list was used as a lens with which to observe the innovation diffusion process within the studied communities, as will be further described in subsequent sections.

In addition to the hacking innovations above, the expert panel also recommended a total of eight potential case study locations. These locations, based on the experiences of the panel, were online hacking communities where numerous hackers participated in discussions of attacks and where a number of the hacking innovations above had emerged. From the initial eight locations, a total of four were selected for further study based on factors including the number of active participants, the length of time the community had been in existence, whether or not the community was open to registration from the public, and a cursory preview of the community to determine if innovation discussion was occurring. The next four sections of this research describe each of the four locations selected for study.

### *4.2.2 – Case Study Analysis - Community A*

#### 4.2.2.1 Community A – Overview

The first community examined, Community A, is one of the better-known underground hacking forums. This community was created in 2007 and has a total of approximately 2,680,000 member accounts that have been registered since the community was founded. It should be noted that many of these member accounts have been closed for abuse by the community staff, for reasons including spamming or breaking the forum rules. One abusive member may have registered numerous accounts that were subsequently banned by the forum staff. Nevertheless, thousands of members appear to be active at any given time – at the time of the data collection of this study, hundreds of users were routinely online at once, and the site boasts that the most users ever logged in at once was over 2,800. Furthermore, approximately 470,000 of these accounts have ever made one or more posts, which for the purposes of this study will be thought of as the active size of this community. Anybody can register for an account on the forum, which is required to access many parts of the site, conduct searches across the forum posts, or send messages to members. Once a user has registered for an account, they appear have access to the entire community.

After a user account was created to facilitate data collection, a comprehensive review of the community was undertaken to understand the community norms and social dynamics. This review revealed that there were numerous types of user accounts arranged in a hierarchy. The more expansive accounts, with greater abilities, required a small payment using an online payment system such as PayPal or Bitcoin. This appeared to have been done not as a moneymaking endeavor but as a way to weed out individuals who wished to conduct abuse on the site. Individual members can also be assigned reputation scores by other members, to help determine which members are trustworthy and which are “scammers” or “rippers”. Members post messages in threads, which other members can reply to in order to have a conversation. Many of these threads have hundreds or even thousands of replies and these conversations can last weeks, months, or even years in some cases. The forum itself is divided into numerous sections, where members can discuss different types of attacks, can post tutorials (or “tuts”, using the forum slang) about certain techniques, or can participate in a marketplace where hacking tools and services are bought and sold. This marketplace confirms what Holt (2013) had described in his own review of hacker communities.

Using the built-in search function for Community A, searches for each of the innovations described by the expert panel were conducted. Coding was conducted on the search results to identify the specific tool or technique being discussed, as well as coding for keywords associated with innovation activity. Examples of coding employed in the data collection and analysis of this case include the grouping of common terms, such as “SQL injection”, “Havij”, “SQLi”, under a single SQL injection category, or “DDoS”, “booter”, “stresser”, “amplification”, “reflection” under a single DDoS category.

Additionally, coding was employed to group postings into various adoption factor or phase categories, by analyzing word choice and meaning to gauge adoption intention. Relevant excerpts from these conversations were placed into word tables, as can be observed in Tables 5 and 6, and for similar tables for each innovation, to assist in comparison and analysis. These word tables allowed for cross-case synthesis to occur following the completion of each case.

The four propositions described earlier in this paper (P1, P2, P3, and P4) were examined and theoretically-derived patterns related to each were obtained from them for pattern matching during the analysis of data from this case. Using these patterns, explanation-building was conducted on the results from each innovation. These explanations were refined for each innovation, and subsequent explanations from later communities reflect and expand upon those explanations derived from earlier communities.

This initial review found that many of the innovations were too commonly-referenced by members and the number of posts requiring review would have been unmanageable (such as “SQL injection” as an overall technique, “booters”, or “crypters”) and as such were unsuitable for the scope of this case; other innovations were barely discussed at all (such as “cross-platform malware” or related innovations). Based on this review, the following four tool and technique innovations were selected for further study:

- Havij
- Amplification attacks
- Bulletproof hosting
- Shodan

For each innovation studied, a detailed review of all postings regarding the innovation was conducted.  Certain details were recorded, such as the date it was initially posted and any relevant quotes regarding the development, diffusion, or adoption of the innovation. The number of postings about the innovation were plotted over time as a visual aid to help determine the level of adoption within the community.  Note that adoption or rejection of technique innovations may differ from adoption or rejection of specific tool innovations.  A technique innovation, such as amplification attacks in Community A, may be implemented in different ways by members and may be incorporated into a number of individual tools.  These individual tools that implement the amplification attack technique may be adopted or rejected by community members, and will be considered, but will not entirely determine, when establishing the community's adoption or rejection of the larger technique.  Individual members' adoption or rejection of tool innovations, such as Havij or Shodan in Community A, has a more direct impact upon the community's overall adoption or rejection of the tool.  This approach will be applied to all innovations studied in this and the other three communities.

In-depth interviews were conducted with two senior forum members.  Note that for this community, the first two members contacted by the researcher elected to participate. These members were selected because they were long-term members, higher in the social hierarchy of the community, who were observed to be posting in one or more conversations related to the four innovations selected above.  This study will refer to them as Members A1 and A2.  Member A1 has been a member of Community A since 2012 and has posted over 3,000 individual postings, and has a moderate number of reputation points.  Member A2 has been a member of Community A since 2010, has posted over 24,000 individual posts, and has a very high number of reputation points. In addition to these interviews, reviews of the artifacts produced by the community, such as the tools offered for download or the websites were the services were available, were also conducted as part of the case study of Community A.

#### 4.2.2.2 Community A – Havij

The first innovation to be analyzed was Havij, a popular SQL injection tool. Structured Query Language (SQL) injection is a common attack technique which involves the sending of SQL commands, often through unsanitized web forms, which allows attackers to enumerate portions of a SQL database (Halfond, Viegas, & Orso, 2006). Postings about SQL injection in general were too numerous within Community A to be feasible for collection for this study; however, postings about Havij were of a quantity that was more manageable. Havij automates much of the manual work of SQL injection and allows attackers to quickly locate desired information, such as the username and password to administrative portions of the targeted website, and is an important initial step in a larger-scale compromise (Parekh, Dave, & Sridaran, 2014). Havij was initially released to the public in July of 2009.

The initial need for the Havij innovation within Community A was observed to be a lack of knowledge about, and difficulty executing, SQL injection (sometimes referred to as “SQLi”) attacks. It was observed that discussions about Havij were broken down into two primary areas: general discussions about the usage, availability, or other questions about the tool; and tutorials authored by members to educate others on how to use the tool. A quantitative analysis of the number of general postings about Havij over time is described in Figure 5, and a similar analysis of the number of tutorials about Havij over time is described in Figure 6.

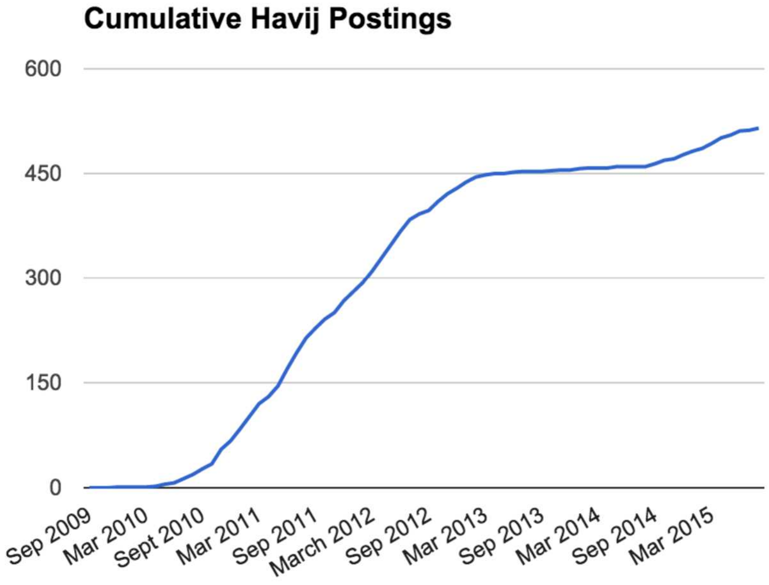


**Figure 5: Havij Non-Tutorial Postings in Community A**

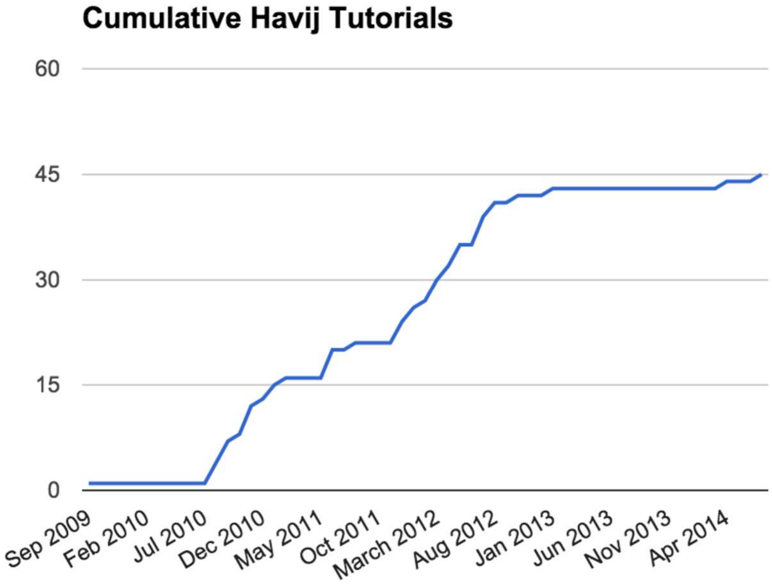


**Figure 6: Havij Tutorials in Community A**

Both curves above resemble an "S" shaped curve, similar to what is predicted by Rogers (2003) for innovation adoption. DOI predicts that the innovators and early adopters are present at the start of the curve, with the majorities in the middle and the laggards near the end. This may not be entirely analogous, as the same population may not be present for the entire adoption cycle as some members may join or leave the community while it occurs; however, the presence of the curve does suggest that an acceleration and then leveling-off of the adoption of the innovation is occurring as it diffuses through the community. Subsequent paragraphs in this section will explore other aspects of DOI to determine if the theory is useful in the understanding of hacking innovation adoption.

Further analysis of specific conversation threads regarding Havij revealed that many adoption factors were present in the conversations. Table 5 describes many of the conversations observed that had aspects of DOI adoption factors.

**Table 5: Havij Innovation Factors in Community A**

| Date | Factor | Sample Quote |
|---|---|---|
| December 2007 | Compatibility | Im skeptical as to how well its mssql injector would work. It could be set back by all sorts of errors, like unclosed braces. |
| | Complexity | If you are going to use a program at least use a nice one like sqlmap http://sqlmap.sourceforge.net/ though I doubt many people here will get to grips with a cli lmao. |
| September 2009 | Observability | wowowowowow this just rocks i got the usernames and passwords for the provided sites now i m gotta search using dorks |
| | Image | This is the problem with you people .. Tool make works easy but it wont give you knowledge .. A Hacker need the knowledge more than Automated tools .. Your not in a competition No Need to Hack Soon .. Even your work is slow if you have knowledge . anyplace in the world you can do SQL Injection without any problem or tools I've been doing SQL Injection for 3 years down ..I never use tools learn the theory of SQL Injection .. how to fix the bug ..Those are what u need |
| May 2010 | Relative Advantage | I'm using this tool now it's very good, I recommend you use it too, automatically SQL Injection always needed and better than manual in some case, Thanks for the tool |
| | Complexity | Look like an automated SQL injector, everything is 'auto' here. |
| | Trialability | Much appreciated mate! I've been reading the tuts and threads posted here.. all of them are awesome. I'm sure I'll be defacing a website quite soon :D. Cheers ;) |
| | Observability | Wow This is great I already got into 2 websites! :) Thanks a lot! |
| | Image | No one who really wants to be a hacker would use this.. the one who uses this willl just get addicted to using fast methods and will alwyas think that hacking is a very fast procedure .. in everything he will loose his patience resulting in him remaining a script kiddie forever.. so if anyone is really interested in being a HACKER (not script kiddie..as the poster himself is) then you should not go for any tools until its just really impossible .. |

Interestingly, the image factor played a significant negative role in the adoption of this tool – using Havij appeared to reduce the user’s social status within the community, not enhance it. As described in the quotes in Table 5, and through a review of the tool itself, Havij is a “shortcut” tool that allows attackers to quickly identify SQL injection flaws in vulnerable websites; however, the same attacks could be done manually if the user possessed the requisite skills. Users of the Havij tool were often derided as “script kiddies”, or “skids” in the slang of Community A, who did not have the knowledge or ability necessary to perform a SQL injection attack by hand. This reduction in image did not appear to slow the overall adoption of the tool, as was observed in the quantitative analysis described above, and many postings were noted where users would state that they use the tool for convenience but that they could have done it manually if they had wanted.

In line with this reduction in image, the two senior members of the community that were interviewed were also similarly derisive of the tool. One of the interviewees, Member A1, stated that Havij is “meant to make even the worst hacker into a ‘Sup3r l33t heker’” but is actually less effective because it “won't deliver the results someone who actually spends time doing manually, as the program cannot change specifics that a human would”. Similarly, the other interviewee, Member A2, stated that within Community A, “it's mostly just people asking how to use it because they think they'll be able to become SQL injection pros very quickly”.

The specific hacking need that this innovation fulfilled was mentioned in many of the above threads. For example, the September 2009 thread – only the second time that Havij was mentioned in Community A – the original poster described the need as “Havij is an advanced MSSQL Injection tool that can check websites for MSSQL vulnerabilities and many other useful tools… Now you have to find a vulnerable website. For example now i'm going to hack this website.” The need that this innovation fulfilled was the hacking of a website through SQL injection. Note that hacking of this type bypasses many traditional IS security countermeasures, including firewalls, antivirus, and IDS. This need existed prior to the introduction of this innovation, but Havij brought SQL injection down to the level where any member could attempt it, and allowed for lesser-skilled members to hack websites that were previously out of their reach.

Many of the postings also appeared to transit the DOI adoption phases of knowledge, persuasion, decision, implementation, and confirmation as members weighed in and contributed to discussions about Havij. Two example discussions that highlight the phases of adoption of Havij within Community A are described in Table 6 below. Note that for the most part, individual comments below are posted by different members.

**Table 6: Havij Adoption Phases in Community A**

| Date | Phase | Sample Quote |
|---|---|---|
| January 2011 | Knowledge | [Initial post] |
| | Persuasion | Should add a DL link to the tut ;) So is this better than SQLi Helper |
| | | Please, please make sure you can manually inject before using Havij. If you don't know and you only use Havij, all I can say is wow. |
| | | Oh dear fuck, the skids get it on a plate now.... |
| | | Not a bad tut, but i personally HATE Havij. |
| | Decision | Manually go through a site and find all of this: 2-3 hours minimum (for me, atleast) Donwload this and run it on the same site: 10-15 minutes |
| | | Good detailed instructions, but I'm not a big fan of Havij... never used it, but I don't plan on using it either! |
| | | I can see now why I read people's posts saying 'I prefer to do it manually.' This takes all the work out of it. Anyone can do this. Well It's a good place for me to start. Great tutorial, easy to follow. I will make use of it. |
| | Implementation | this tool is pretty awesome but i think you should know how you can do this stuff without tools ;) |
| | | I got some website shopadmin tables, and in order column I can't get the number of credit cards. It only shows me **4251 . I mean only the lasts 4 digits. What can be done? Is this because of havij free and dont let me see the all cc number? Or because some websites encripts credit cards detail? If so, what can be done to unencrypt them. Thanks. |
| | Confirmation | thanks a lot, i hacked my first website with this, altho now i prefer manually |
| | | Nice Im using it Thansk for this stuff. It really awesume. |
| | | Prefer to do it manually, but this will do the job faster :) |
| March 2012 | Knowledge | [Initial post] |
| | Persuasion | Nice tutorial, have used them before but.. Better learn Manual SQLi .. |
| | | Havij is a pretty straight forward program very nice n i cant wait to try it |
| | | This is a nice entry into the realm of sql injection. |
| | | YA GOOD TUTORIAL BUT ITS THE LAZY WAY OF INJECTION.. |
| | Decision | nice tutorial brother..!! willl try this out :D |
| | | good tutorial going to give this a try i have havij havent used it yet |
| | Implementation | How many tries does it take to hack 1 website? Because I have tried about 30 sites now and all of them I couldn't find the admin page or admin credentials or i did find them but they just didn't work. |

Knowledge of Havij was provided through the initial posting by the community member who was sharing the post – in both cases above, these postings were tutorials intended to promote the user of the tool. The next phase, persuasion, occurred as other members weighed in on the tool and expressed their opinions, and attempted to influence the favorable or unfavorable attitude of potential adopters. In many cases, the image factor played heavily into this phase, as detractors of Havij claimed that the use of the tool would reduce the user's social standing in the community, through such comments as "it's the lazy way of injection" or "the skids get it on a plate now". In the decision phase, members would state if they decided to adopt or reject the innovation, as can be seen through quotes such as "going to give this a try" or "I don't plan on using it either". In the implementation phase, members talked about their experience using the tool, both positive and negative ("I have tried about 30 sites now and all of them I couldn't find the admin page"). Finally, in the confirmation page, the members declared their intention to continue to use the tool ("Nice Im using it Thansk for this stuff. It really awesome"), or discontinue use ("thanks a lot, i hacked my first website with this, altho now i prefer manually").

Because of the reduction in image brought about by using the tool, it is also possible that some of the members may be moderating their publicized adoption of the tool. That is, these members may be claiming to not need the tool while in reality adopting it for use, perhaps covertly. Many of the conversations observed within Community A regarding Havij, such as the quotes referenced above ("i personally HATE Havij", "this tool is pretty awesome but i think you should know how you can do this stuff without tools", and "thanks a lot, i hacked my first website with this, altho now i prefer manually") lend credence to this notion that this innovation may be adopted in secret to avoid being labeled as a "script kiddie". A review of DOI literature has indicated that this may be a unique phenomenon within innovation studies. Certainly, some innovations may be adopted in non-public ways, such as criminal innovations both cyber and real-world (Rogers, 2003); however, moderating visible adoption of an otherwise useful innovation to preserve image within the social structure appears to be a new finding.

While only two sample threads were described in Table 6, numerous other threads were observed where the phases of adoption were traversed by the members of Community A in regards to Havij, indicating that the pattern observed above is representative of overall adoption characteristics within this community. As a whole, it appears that Havij was adopted by Community A.

#### 4.2.2.3 Community A – Amplification Attacks

The second innovation studied within Community A was the amplification attack, also known as the reflection attack. Amplification attacks are a type of denial of service (DoS) attack where spoofed network packets are sent to vulnerable servers, which respond to the intended target with a much larger packet in response. A common type of amplification attack is DNS Amplification, which has a typical amplification factor of 14, meaning that the target receives a DNS packet 14 times larger than the spoofed packet sent by the attacker (Anagnostopoulos, et al., 2013). Other types of amplification attacks include NTP, SNMP, CharGen, gaming protocols, and peer-to-peer filesharing protocols

(Rossow, 2014). Amplification attacks are widely used in booters or other DoS-as-a-service offerings (Karami & McCoy, 2013), as described by the expert panel above.

Unlike the discussions about Havij, discussions about amplification attacks in Community A were much less oriented towards instruction and education. Indeed, only a total of four tutorials were observed regarding these attacks. Instead, many of the postings were threads related to the sale or discussion of DoS code or services that utilize the amplification method. A quantitative analysis of the total number of amplification postings was also conducted, as described in Figure 7:

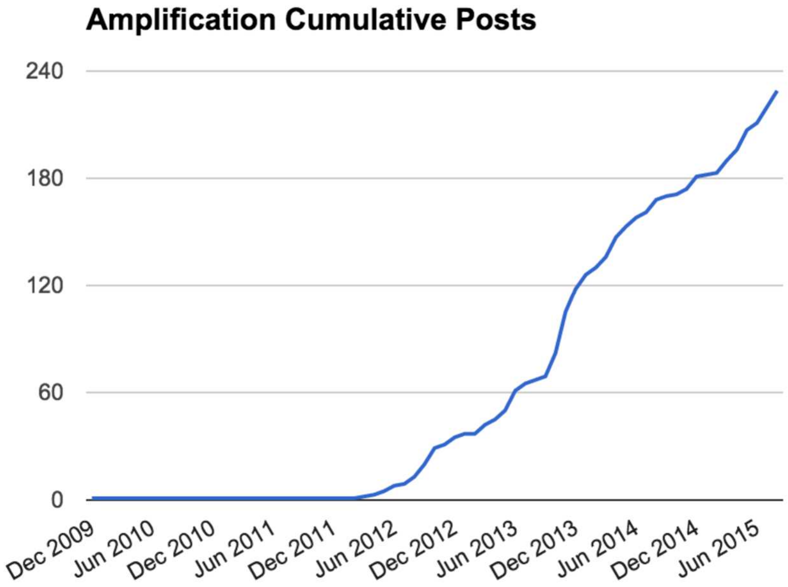


**Figure 7: Amplification Postings in Community A**

The curve in this example is less clearly of an "S" shape than the Havij curves – Community A had a single post about amplification attacks in 2009, and then nothing until March of 2012, when the technique began to pick up steam. Instead of leveling off as the technique became widely adopted, it continues to rise, even at the time that data collection within Community A had ended, which may indicate that it has not yet been fully adopted. The “S” curve model may also require modification in an environment where many of the posts are about sales about tools or services. It was noted that many of the tools and services offering amplification attacks – such as booters – would be initially offered, garner a significant amount of attention in the community, and then eventually go offline as either the operators moved on or the services were shut down, perhaps by network operators who observed the abusive traffic or even by law enforcement. New services would be offered by different sellers to take the place of the earlier services, a cycle the appeared to continue indefinitely, meaning that estimating who the late majority or laggards were in this situation could be difficult. From the initial posting in 2009 – one of the first innovators of this technology – and then to the early adopters observed in 2012, this innovation appears to have been successfully adopted by Community A.

Similar to the Havij innovation, many of the threads regarding the amplification attack technique could be broken down into individual DOI factors. Table 7, below, describes a few example threads where this can be observed.

**Table 7: Amplification Innovation Factors in Community A**

| Date | Factor | Sample Quote |
|---|---|---|
| April 2012 | Relative Advantage | Hey guys, Over the past few days, I have figured out a few new unseen amplification methods and would like to get your opinions on it. Some of you may think that it it is only possible to amplify off of game servers although there are many other possibilities. Here is an example of one of the couple methods that recently found and created. I would just like to get some of your opinions on it. Thanks. |
| | | This is not a using the old quake method, The quake method only has 4x amplification at max, This method as you can see has about 24x |
| | Trialability | Yea, if someone would like to donate a 10gbps server then we could fully test the power of this method but as it stands we are downing a 1gbps server with 4MB |
| | Compatibility | Man that will make one hell of booter. Would make it cheap as shit to maintain one now instead of having multiple you would only need like 2 |
| | Complexity | It's not just game servers that are being utilized, that's simply the most common here. They're very easy to manipulate. You literally just send chr255chr255chr255chr255getstatus\n in a UDP packet to the gameserver, and it replies back with the map being played, the players that are currently on, and some other shit. Easy as hell. |
| | Observability | OP, if your method is truly 24x amplification or whatever you claim, why is WickD's booter not outputting 6x what it used to? |
| | | I tested this with LiteSpeed a few minutes ago and it peaked at 6gbps. That sir, is fucking amazing. |
| May 2015 | Relative Advantage | I have access to a very fast server and I've been scanning for amplification lists. So I figured I would release them right now for everyone, along with their scripts. Enjoy Black Hat I am able to scan 1.1.1.1 to 255.255.255.255 and it will only take 2 hours. |
| | | Completely maxed 1gbit port after filtering the list. It killed the box after the 104M incoming so i had to stop it. |
| | Compatibility | thanks again man so the netbios method will work for protected hosts, home connections, w/e? layer 4? |
| | Complexity | ya, most home connections can be downed with just 300mpbs from anything, its nothing really, just another amplification method that can be used. not exactly specific to anything since it will still push data at anything, only a matter of time before ovh, nfo, bl filters this out too, if they dont already. |
| | Trialability | I used the NTP monlist script that comes with Nmap (nmap -sU -pU:123 -Pn -n --script=ntp-monlist <target>) but I changed the command to "nmap -iL ntpserverslistfromroot.txt -sU -pU:123 -Pn -n --script=ntp-monlist". For any other skids the -iL arg is inputList and the list in question is the one that Root gave us in the last page (big props to Root for that :D Thanks!). Here is the list in Pastebin (I am a new user so I can't post links yet :/ Replace the [DOT] with a "." (no shit) and the [FWSLASH] with a "/". |
| | Observability | did you not see my dstat? snmp pushed 10gbps from one box, that hardly sux0rz. Anyway, it's just another amp method so ofcourse it's gonna be blocked/firewalled. |

In the postings above, the various innovation factors can be observed as the community learns about the new amplification techniques. Most reactions to the innovation were positive and indicated that members were likely to adopt. Conversations regarding this innovation did not have explicit mention of the image factor – however, in the above threads as well as numerous other threads related to amplification, appreciative members would either "vouch" for the services of the poster who was offering the amplification service or tool for sale, or would give "rep" or reputation points as a way of enhancing the poster's status in the community.

The hacking need was described in many threads related to this innovation. Members of Community A desired to use DDoS attacks against websites to take them down, but waging these attacks was difficult due to the large number of bots required to send the traffic. Websites had been increasingly adding bandwidth and security countermeasures to defend against traditional bot-based DDoS attacks. Amplification attacks allowed members to use far fewer attacking machines, with greater overall bandwidth, to take down a victim site. The need for this innovation was pre-existing – the desire to conduct denial-of-service attacks – but when amplification was introduced, they became much easier.

It was also observed that many of the threads related to amplification attacks transited the innovation phases. Table 8 describes one such thread, involving a new DoS service available for rental utilizing amplification attacks, where the innovation phases were clearly observed. Note that names of users or websites, when observed within relevant quotes, were removed by replacing the name with the text "[redacted]". This will be employed for any similar occurrences that are observed elsewhere in this study.

**Table 8: Amplification Adoption Phases in Community A**

| Date | Phase | Sample Quote |
|---|---|---|
| July 2012 | Knowledge | [Initial post] |
| | Persuasion | [redacted] has my vouch. Haven't talked to him much, but I can definitely say that whatever he produces is legit. |
| | | I am currently in the Process of buying this its So fuckin badass with all the features and what it can do and its cheaper :) and more power Vouch |
| | | Vouch for this, great booter with loads of power. GLWS. |
| | | [redacted], is there a reason everytime someone says an amp even close to what you claim [redacted]'s is, you have to whine about it for the first 3 pages of their thread? I know we had some details wrong on stresser++, but you do realize that this is the exact same thing as what I'm using, and you stopped posting about it... Also, I'd like to point out with [redacted]s "500mbit" plan, it hits 34mb, about 272mbit (I have a copy, I've tried it.), so I don't see how you can keep complaining about what we're saying when your effectively cheating all of your users. |
| | Decision | Probally gonna try this once my [redacted] licence end. Less expensive and more powerfull. :P |
| | | please add me on aim…[redacted] really interested in purchasing… |
| | | So the main question is..how much are you writing without usleep? |
| | Implementation | we just bought This Sick AF API its exactly as powerful as they say it is :) |
| | Confirmation | Yes, first it was totally offline and after a while it did hit with 33M again and now idk since i havent tested again but hes not even replying to anyone though seems he just ignores customers when things go wrong. |
| | | API is still not working. Fix your shit all I am making a dispute. Hasn't worked for 1 day yet. |
| | | Gamma API is currently offline. We are working on the issue and it should be resolved today. |

In the above example, a phenomenon was observed that regularly occurred in Community A. The tool was initially announced to the community by the sellers, which consisted of junior-level members who acted as the sales staff or technical support for the tool, and a senior-level member who was the ultimate author of the tool. These individuals are the innovators and the champions of this tool, and this announcement served as the knowledge phase for this innovation. Initial reaction to the service was skeptical, but then more experienced members chimed into the thread with vouches for the service or for the author in the persuasion phase. These senior members acted as change agents from a DOI perspective, and their vouches provided community members with assurances that the offering was legitimate, and led to requests from members to the sellers to purchase access to the service.  Then, also in the persuasion phase, a champion of a rival service added disparaging remarks to the thread, doubting the technical abilities of the tool.  Later, additional senior members provide additional assurances, and in the decision phase members appear to be interested in learning more about the service or stating intent to purchase.  In the implementation phase, members stated that they were using the tool.  However, in the confirmation phase, those members began complaining that the tool went offline and that the sellers were unreachable.  In the end, this implementation of amplification attacks by the service sellers was not adopted by the community, even though amplification attacks as a technique were enthusiastically adopted.

A similar theme of senior members acting as change agents and lending support to tool innovations was observed in another amplification thread.  In this thread – which also occurred in early 2012, shortly after the amplification technique began picking up steam – a service was offered by a seller and disparaging remarks were made by other members: "You mean only enough power to take down wendys? Lol. Wendys is the weakest of them all. And how do you not have enough time to do something that would take a total of about 2 minutes?", and "You continue to browse and make post and shit but can't do a simple request that takes 10 seconds? Exactly. Proves you can't down any of them other than Wendys. I take my vouch back."  However, a senior member with significant reputation points and a higher-status account defended the seller, in a fashion: "Omfg guys i don't like [redacted] but stop bitching on his thread you've never fucking tried to booter so fuck off and go fuck around another place its pissing me off i see you [reacted] on any booter threads and bitching about them when your booter aint even half as strong as this." After this defense by this respected member, other members began purchasing the tool and remarking about its abilities: "Thanks, this booter does hit hard.. It has 31 fucking vps's that can each hit home connections & websites down, so 1x31, do the math people!" and "Looks like a great booter. GLWS broskie :P", for example. These examples indicate that in Community A, the ability of senior members to act as change agents is an important part of the decision to adopt the innovation.

The two senior community members, Members A1 and A2, were also interviewed about amplification attacks. Member A1 stated that amplification attacks were extremely popular within Community A and were adopted quickly because the technique was a "much better method of taking down web servers and/or home connections". Member A1 also stated that he had "utter disdain for DDoSing" and as a result had an incomplete understanding of the technical implementation of the attack, but had observed its enthusiastic adoption within the community. Member A1 related that the amplification technique was originally championed within Community A by Member A2, and as a result made a significant amount of money by selling tools and services.

Member A2, the original change agent for the technique within Community A, was interviewed and described the process of adopting amplification attacks, stating that they were adopted so quickly because they were far superior to the previous popular DDoS method for booter services, which required hundreds or thousands of "shells" be installed on compromised systems to generate UDP floods for the attack. Regarding trialibility and observability, Member A2 stated the following:

> *Back then a lot of new customers would just request "demo hits". Sellers would provide them free of charge for a few minutes so the buyer could see what he was getting. Sellers of these services also posted a lot of http://downforeveryoneorjustme.com links showing that they took down large or popular websites. That was enough demonstration to convince everyone. After that, word of mouth spread it quick.*

Member A2 explained that implementing the amplification attack technique within booters allowed Community A members to rapidly experiment with the technique. After the first booter implemented amplification attacks, Member A2 stated that other booter operators reacted quickly: “After people saw the results of the first booter running it, every seller raced to get the scripts needed and implement it into their booters.” As a result of the integration within the booters, Member A2 recalled that “it caught on quick” and “it very quickly became the norm”. This is an example of *reinvention* of the amplification technique into the booter operators’ own services.

#### 4.2.2.4 Community A – Bulletproof Hosting

The third innovation studied within Community A was the technique of “bulletproof hosting”. Bulletproof hosting refers to the hosting of websites or domain names upon hosting providers that will not take down illegal or malicious content, even if a complaint is made, due to either geographic location in a country where computer crime laws are weak (Wei, Sprague, Warner, & Skjellum, 2010) or because the operators simply offer such services because they are profitable (Kamluk, 2009). Member A1 defined it as “hosting that will not go down, and typically allows for malicious websites, exploit kits, etc. This hosting should have the capibility to withstand large amounts of traffic, and DDoS protection as well." Bulletproof hosting remains popular, even though a number of early high-profile services were shut down following media attention and government investigation (Bleaken, 2010; Clayton, 2009).

Within Community A, a number of discussions regarding bulletproof hosting services were observed. These postings consisted of a range of topics including questions about the technique itself, where to procure the services, and members offering bulletproof hosting services. There were no tutorials offered by members, likely because operating a bulletproof hosting service requires a significant amount of resources and community members appeared more likely to purchase services from one of the other members who were serving as vendors. The quantitative analysis for bulletproof hosting in Community A is represented in Figure 8:

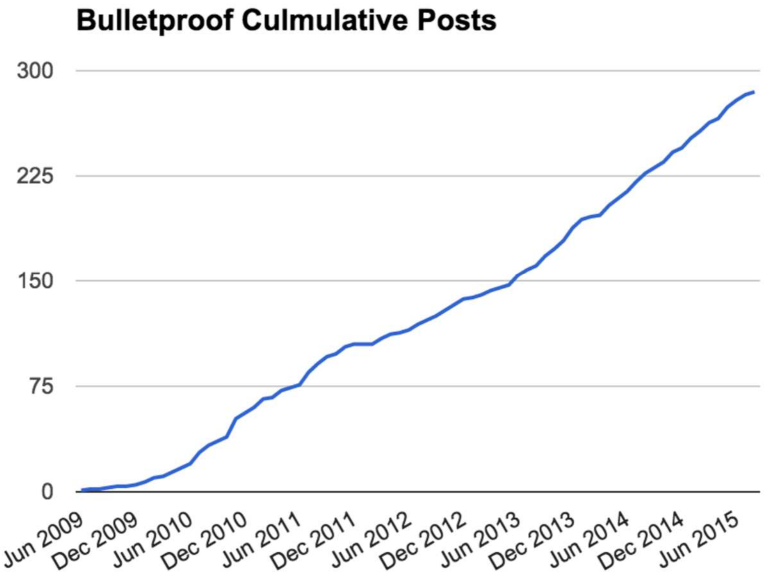


**Figure 8: Bulletproof Hosting Postings in Community A**

Unlike the previous two innovations examined for Community A, this curve is much more linear.  This may be because, similar to what was observed for amplification attacks, many of the postings in Community A related to bulletproof hosting were centered around the sale of services.  This technique is one that is required for many modern computer crime operations, including botnets, spam, banking Trojans, and the like (Bleaken, 2010; Wei, Sprague, Warner, & Skjellum, 2010); given this constant need, it is reasonable to assume that vendors will continue to post advertisements for services. Indeed, if we remove postings related to the sale of such services from the analysis in Figure 8, we obtain the results in Figure 9, which is still somewhat linear but does resemble an "S" curve more than the chart with commercial postings.

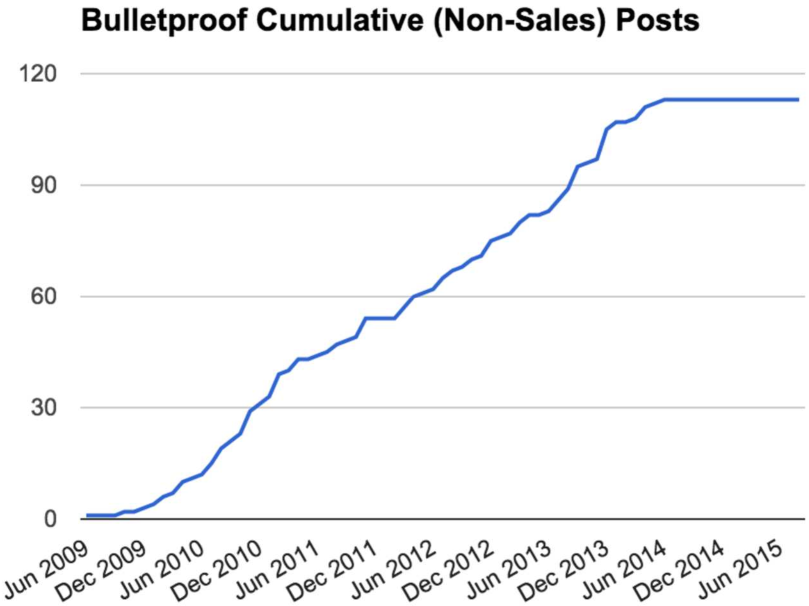


**Figure 9: Bulletproof Non-Commercial Hosting Postings in Community A**

A review of the postings related to bulletproof hosting revealed discussions centered around a number of DOI factors. Table 9, below, describes a few example threads where these factors can be observed.

**Table 9: Bulletproof Hosting Innovation Factors in Community A**

| Date | Factor | Sample Quote |
|---|---|---|
| June 2010 | Relative Advantage | I will be setting up bulletproof dns's that will work as a no-ip, but won't get canceled. … Bulletproof is great for anyone running a botnet/trojan. No-Ip.com cancels domains if any abuse report is received, with these sub-domains/Dns redirects you will be guaranteed to have them working all the time!<br>Instead of having a dns like happy.no-ip.com for your bots to connect through, I offer Bulletproof dns's happy.mysite.com and they won't get canceled due to abuse. |
| | Complexity | In order for you to setup a service like this you need multiple dedicated servers...which I doubt you will be able to invest in. Anyway good luck. |
| | Trialability | Can you pm me an ip of your dns so I can check location etc? I might be interested |
| May 2011 | Relative Advantage | I am selling 100% bulletproof shared, vps, and dedicated hosting in Zimbabwe. EVERY content is allowed, with the following restrictions: No email spam or resource eating script on shared/vps, you must get a dedicated server for this. Hate speech against the government of zimbabwe is not allowed on ANY service. |
| | | Nigeria is not fully bulletproof. Due to the state of the government in zimbabwe you can host absolutely anything on this and you will never, ever be shut down. |
| | Compatability | So which hosting should I take so that I may host web booter and private shells there with mass mailing feature...? |
| | | when you mean anything you really mean anything ? if you allow anything prices are big but should worth it. |
| | Observability | Guys, I have the dedicated server from this guy. Its quite expensive, takes up a lot of my pay-check but its still worth it. The speeds are fabulous, customer support is beyond amazing, and you can do w/e you want with it. I highly vouch for this member. |

Unlike the previous two innovations in Community A, it appeared that most members intrinsically understood the necessity for the bulletproof hosting innovation. When members sought knowledge, instead of asking questions about the technique itself, members asked for places where they could purchase the service. Most threads were not about raising awareness or convincing members to adopt the innovation; instead, most threads appeared to be about selling services. This demonstrates that the hacking need was fully in place prior to the introduction of this innovation – members needed places to host their criminal infrastructure without fear that they would be shut down before, during, or after their attacks.

The very first posting related to bulletproof hosting within Community A, observed in June of 2009, was a request for help finding a reputable service: “But the thing is when your bot is reasonable big it gets reported and shut down. You'll end up losing all your bots at once.  So is there any service out there that offers bulletproof dns hosting? so even if they get reported they dont shut it down. A service that is known to host botnets?”. Another early posting, from October 2009, also was a request for finding a hosting provider: “Hi, can anyone advise on a good remote service to host my own irc server. I want to experiment with bots a bit more. I've looked into vpn's rdp etc I'm willing to pay something for it and was looking at possibly something off-shore I know many countries have very limited enforcement against botnets. Basically does anyone host their IRC like this and where is a good place to arrange it?”.  The analysis of this innovation within Community A revealed that the primary decision was not to adopt or reject the innovation; if members knew enough to ask about bulletproof hosting, they already were prepared to adopt it.  Instead, the primary decision was with which provider of this service to adopt.

In that light, there were a number of sales-oriented threads where the innovation-decision process about bulletproof hosting providers manifested clearly.  The below table describes a few example threads that detail this process.

**Table 10: Bulletproof Hosting Adoption Phases in Community A**

| Date | Phase | Sample Quote |
|---|---|---|
| September 2015 | Knowledge | [initial post] |
| | Persuasion | Big vouch, op hosting gr8 seller |
| | | Vouch for [redacted] both legit members, AnonTech is amazing bulletproof hosting, Don't hesitate to buy. |
| | Decision | Any vouch copies? Would be interested in writing a review. |
| | | mass mailing? phishers? botnets? Just need clarification |
| | | The bandwidth is a joke to be honest. |
| | | Say, would I be able to host a botnet that will include DDoS? Or is this against T.O.S.? Thanks :) |
| | | What is your definition of bulletproof? I know for a fact if you get a spamhaus complaint it will be shutdown fast. |
| | Implementation | Huge vouch! Everything worked perfectly. :) |
| | | got my first purchase few min ago. swiftly all works well |
| | Confirmation | Well, just bought anon.tech hosting with domain, fast, kind and helpful support, fast hosting with good price, had no problems, I reccommend this to buy |
| June 2015 | Knowledge | [initial post] |
| | Persuasion | high price man ... it,s about 7 or 8$ enough anyway best of luck for you sell |
| | | Amazing bulletproof hosting, vouch for [redacted].. always comes with HQ products. A+ to support |
| | Decision | Can i use it to send out like 10,000 emails per day? "marketing"... |
| | | DMCA ignored!! Really nice... will like to test it out soon! |
| | | we, usually won't get any spam report. If we got any we change the ip. let me know if you do have any other query, thanks |
| | | There is no atomatic layer 7 protection, we do offer basic ddos protection upto 120gbps as per packet flow size. |
| | Implementation | Ordered this hosting, works how it belongs to 10/10 + great support ^^. |
| | | Had a little problem, but it was fixed fast. <3 VOUCH! |
| | | I believe this works well for me am giving it a try right now. |
| | Confirmation | VOuch .. Still happy with the host i am loving it :) |
| | | i have told you many time bro but you keep deaf hear of that, pls fix the two website Asap is upto 2weeks now or is it when it will expire that you will start fixing ? |
| | | Op calls bulletproof hosting because he will allow you to host whatever you want. I used before his host, even got a dedicated IP and at first report, i was suspended lol |

As can be seen above, individual members made decisions to adopt or reject this seller's innovation, with a majority of those who contributed appearing to adopt. Some members in the confirmation phase appeared to reject the service after purchasing – such as the member who complained about being suspended at the first complaint, and the member who complained about a problem that persisted for two weeks – but the complaints of these members did not appear to detract from the enthusiasm of the other members participating in the thread. Like the previous amplification innovation, it was observed that in the persuasion phase members would supported adoption would provide vouches for the seller, while members who did not encourage adoption would contribute disparaging remarks. In the decision phase, technical questions were posed by members and were answered by the seller. In the implementation phase, members would describe their experience with the service, often adding additional vouches to encourage further adoption, and in the confirmation phase, members would highlight their intention to continue to utilize the service. It appears that the bulletproof hosting technique was indeed adopted within Community A.

Members A1 and A2 did not have much experience with bulletproof hosting services, and stated that Community A is not the type of community where true bulletproof hosting can be bought or sold due to relatively low sophistication of average members. Member A2 stated that many sellers of such services “claim their shit is bulletproof but it isn't. Not here at least”, and added that “not many people here actually know how to obtain bulletproof hosting nor do they want to pay the costs required. Those that do are usually more active on other sites.” Even if the quality of bulletproof services offered by sellers within Community A was poor, there was still a level of interest in members seeking to adopt services, and many of the services offered by sellers were experimented with by community members before ultimate adoption.

### 4.2.2.5 Community A – Shodan

The final innovation recommended by the expert panel and observed within Community A was Shodan. Shodan is a search engine that scans the internet and indexes administrative pages of internet-connected hardware devices, including so-called "Internet of Things" devices like surveillance cameras (Genge & Enachescu, 2015) or industrial control systems such as manufacturing or power generation devices (Bodenheim, Butts, Dunlap, & Mullins, 2014). Hackers have utilized Shodan to maliciously attack baby monitors, webcams, heating and cooling systems, and other connected devices (Hill, 2013).

A significant amount of mischief can be conducted using these connected devices. As a result, there was interest in this innovation from the members of Community A, although not to the levels that were observed for the previous innovations. The chart of postings over time is displayed in Figure 10:

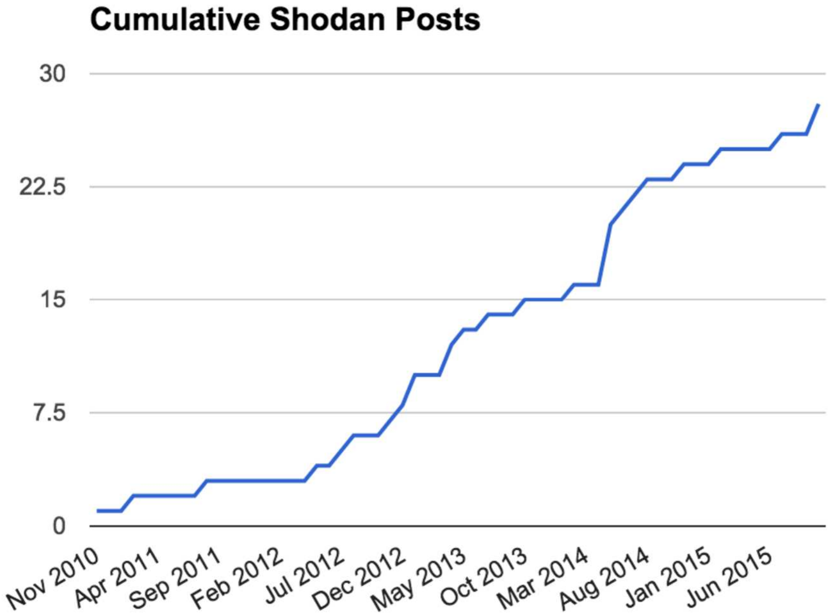


**Figure 10: Shodan Postings in Community A**

For this innovation, there were only 28 observed postings referencing Shodan, indicating a relatively low level of adoption within Community A.  Neither Member A1 nor Member A2 had any experience with this tool.  Of the 28 observed postings, three were tutorials, and the rest consisted of questions regarding the service.  Due to the smaller number of postings, there were fewer samples of DOI innovation factors observable in the postings regarding this innovation; however, a few examples are described in Table 11.

**Table 11: Shodan Innovation Factors in Community A**

| Date | Factor | Sample Quote |
|---|---|---|
| December 2010 | Relative Advantage | So, I'm going to provide you chaps with a short guide to a very useful resource indeed. It's called Shodan and it functions, essentially, as a port scanner/banner grabber database and search engine…. Well, take a look http://www.shodanhq.com/browse/ there. These are the most popularly rated searches, and include webcams, routers, embedded devices, and more. You can use this to find specific combinations of servers and operating systems in specific countries using search filters, and of course, like all search engines, basic operators will work, too, so you can engineer some fairly damn specific searches. |
| | Compatibility | Shodan contains an exploit search engine which allows you to get results from those sites, yes, but its primary purpose is as a database of port-scans with banners, a search engine of vulnerable servers, which is completely different to a collection of exploits. Did you even read my post or visit the site? |
| | Trialability | Unfortunately, some of the service filters are pay-only (telnet and https), and without paying, you can only see the first 50 results in a search, which is annoying, but you can survive. |
| | Observability | It's a very interesting tool to check, for example, for anonymous ftp. http://www.shodanhq.com/?q=port%3A21+%22...n%22++-530 Or perhaps you'd like to see Windows 2000 boxes running Apache? http://www.shodanhq.com/?q=os%3Awindows2000+apache Uses are pretty much endless; there's plenty more you can do, and plenty more useful machines, so I'll leave you guys to find them out. |
| December 2012 | Relative Advantage | A webcam exploit tutorial that I have used for the popular leak of the strip club cam I found about a month back. I still use this method today and have great success with it and now it is time I share a good HQ tutorial with my fellow HF members.My old thread where I released the popular strip club cam can be seen and located here. |
| | Compatibility | Just as a little side-note. If you find something interesting, but without the default password for that piece of software - most of these are implemented without any kind of brute-force counter-measures. Oh. And I find most user/pass comboes to be admin/admin <-- Depends on what software you target of course. |
| | | "The only thing I knew about the website is that people used it for WebDAV shells. .xmls." "Yes allot of people use it to find shells.I did myself for a bit but exporting xml files can get expensive and most of them are overused." |
| | Complexity | nice tutorial but somehow I cant log into any Cams I think I tryed 30 different cams already but it always says "http://118.80.12.16 verlangt einen Benutzernamen und ein Passwort. Ausgabe der Website: "VIPCAM" maybe they just changed the default settings and its a lot of work 2 finally find one with default settings ^^ |
| | Trialability | "really can u give me a IP that works? iam trying 10 min and got nothing yet :(" "No prob man I have sent you a list of them to play with." |
| | | Lol I must be retarded it's not letting me login it comes with login and password what do I put in.. |
| | Observability | works, funny stuff. thanks for that. I have a lot of successful logins where the cam does not load. Any suggestions? |

Both of the threads referenced above were tutorials, posted by two Community A members to attempt to persuade other members to utilize the tool. These two members can be viewed as change agents who are enlisting other members in the use of this service. The relative advantage of Shodan is immediately apparent to potential adopters, and the trialability and observability of the tool makes for easy testing. The tool was somewhat complex, resulting in numerous members posting questions on how to utilize it properly; however, most members appeared eager to adopt once they had tried the tool.

From a hacking need perspective, this innovation differs from the previous three innovations studied within Community A. Members who participated in discussions about this tool were surprised at the level of information they could obtain from it, and few prior discussions related to a desire to hack webcams or other devices were observed prior to its introduction. The results appear to have been treated as a novelty but did not appear to be a core part of the attacks perpetrated by the members of Community A. This indicates that the tool was introduced to Community A before a legitimate hacking need existed for it, contrary to the need that had been observed for the prior three innovations.

The last conversation in Table 11 was also reviewed for indications regarding the steps of the innovation-decision phases, and the results are described in Table 12:

**Table 12: Shodan Adoption Phases in Community A**

| Date | Phase | Sample Quote |
|---|---|---|
| December 2012 | Knowledge | [initial post – tutorial] |
| | Persuasion | Very nice tutorial. Very HQ for beginners such as myself. Thanks! |
| | Decision | im gonna check this out it seems cool, thanks for the tut |
| | | looks legit man, i'll have to try this out (for education purposes of course ;) |
| | Implementation | Vouch, i am in a cam of a bank :D Nice tut !!! |
| | | Looks nice but, I'm having issues getting the login to work. I've tried quiet a few webcams but, none seem to work. |
| | | Just tried the method out, its real cool. Thanks. :) |
| | | Good post man! Hacked into my local banks camera! |
| | Confirmation | I have bought the extended/full search. Thanks for a great guide! |
| | | Okay, I got it working. Just make sure to try a few, It took like 10-5 for me to get a working one. Awesome method, Vouch. |

Unlike many of the previous innovations discussed, there appeared to be little to no opposition to Shodan. This may have been because there were no competing services being offered by other members, and use of the tool did not reduce the social standing of the user, such as with Havij. There were some members who experienced problems, but the change agent who posted the tutorial provided guidance to these members and appeared to resolve their problems in a way that convinced them to adopt. Despite the low number of postings regarding Shodan, it does appear that this innovation was adopted by members of Community A.

### 4.2.2.6 Community A – Summary

Community A was an extremely active hacking community where numerous innovations were proposed, experimented with, and ultimately adopted or rejected by members. The focus of the community appeared to primarily cater to members who were engaged in the *operations* of attacks and members who were operating as vendors *selling* hacking services, although some education of members who wished to *learn* techniques occurred as well. Many more innovations were observed being discussed and evaluated than the four examined here. None of the innovations examined as part of this case study were actually developed within Community A, with the exception of amplification attacks which were introduced by Member A2; instead, the techniques or tools were introduced to the community by change agents who championed their use or who authored tutorials instructing other members on their use, and many of the techniques (such as bulletproof hosting or amplification attacks) were incorporated, or reinvented, within service offerings by vendor members. Also important to the adoption of these innovations were the senior members who lent their support to the change agents and who influenced the more junior members to experiment with, and eventually adopt, the innovations.

This forum appeared to contain a mix of skill levels, including some lower skilled individuals and some more highly skilled individuals, but the overall skill level of this community appeared to be medium. Of the four innovations examined, only one - Shodan – appeared to have been introduced without a hacking need firmly in place. The other three innovations appear to have been introduced as a response to a need by members to engage in easier and more effective attacks.

The communication of the individual innovations primarily occurred over the channels offered within the community itself – namely, public forum postings and private messages. Some additional communication occurred outside of the community, such as when vendors posted links to their websites where hacking services were offered. Some other communication may have occurred over instant messaging services, such as Skype or MSN Messenger, as many members posted their contact information in their post signatures and invited individuals to contact them there. It was not possible to observe communications that occurred in non-public spaces, but it also appears based upon the flow of conversations within message postings that the majority of communications occurred within individual message threads.

The four innovations that were examined were instructive for an understanding of the innovation process within Community A. These innovations demonstrated the rate of adoption within the community, highlighted the differences in measuring adoption of tools and techniques without a commercial aspect versus ones that involve the sale of services, and illustrated the phases of innovation factors and innovation-decision process experienced by members who were deciding whether or not to adopt as the innovation diffused throughout Community A. Interestingly, the importance of the image factor in the adoption process was also demonstrated, as was the lack of importance of voluntariness. These findings will be compared against the findings from the second case study location, Community B.

### *4.2.3 Case Study Analysis – Community B*

#### 4.2.3.1 Community B – Overview

The second hacking community studied, Community B, was a smaller and seemingly more professional community than what was observed in Community A. The community appeared to be more focused upon education around hacking techniques, versus the focus on the sale of services seen in Community A, with members appearing to have more experience and with fewer "script kiddies". The community was started in 2010 and boasts a total of approximately 38,400 individual member accounts. Like what was observed in Community A, many of these accounts were never used or appear to have been blocked by forum staff for abusive behavior. Many other accounts appear to have been used by members to “lurk” but never contribute to public conversations. A total of approximately 3,700 of the total user accounts engaged in conversations and can be thought of the number of active users of the community. Once a user has registered for an account, they appear have access to the entire community.

Similar to Community A, Community B also had a hierarchy of members, with the forum owner and administrators at the top, followed by moderators, and then various types of “VIP” members, with the general members at the bottom. VIP status is given to members based on merit and participation and is not available for purchase as in Community A. More senior members who have been active for a longer period of time appear to have been awarded VIP or higher status. This community also employs a system of reputation points. These points appeared to be less important for Community B, however, as there were no commercial postings within the community and using reputation to determine scammers and rippers was less important.

Based upon the input from the expert panel described previously in this chapter, a review of innovation activity within Community B was conducted. Using the search tools built into the community’s forum-based communication system, Community B was searched for postings that related to the innovations suggested by the panelists. This review resulted in the selection of four innovations to be analyzed within Community B, as they were found to have a total volume of postings sufficient for study. These innovations are broader subjects then those explored within Community A, due to the smaller size of the community and lack of commercial postings, and can be found below:

- SQL injection attacks

- DDoS attacks
- Crypting services
- Bitcoin (criminal aspects)

Data collection and analysis for these innovations and this community was identical to the method employed for Community A. In addition to data obtained from observing postings about the innovations, interviews with two senior members of Community B, Member B1 and Member B2, were conducted. These members had expressed interest in participation after they the researcher posted a request for participants in a chat room created for the sole use of Community B members. Member B1 is an administrator of Community B and has been active since 2010, having posted approximately 6,500 individual postings, and has a high number of reputation points. Member B2 is a moderator of the Community and has been active since 2011, with approximately 1,100 postings and a similar number of reputation points to Member B1. Both members are longtime senior members with enough visibility to have input regarding the innovations under study.

#### 4.2.3.2 Community B – SQL Injection Attacks

The first innovation studied within Community B was the technique of SQL injection. SQL injection is the hacking technique behind the more user-friendly tool Havij described in the previous community. A quantitative analysis of the number of postings relating to SQL injection attacks can be found in Figure 11.

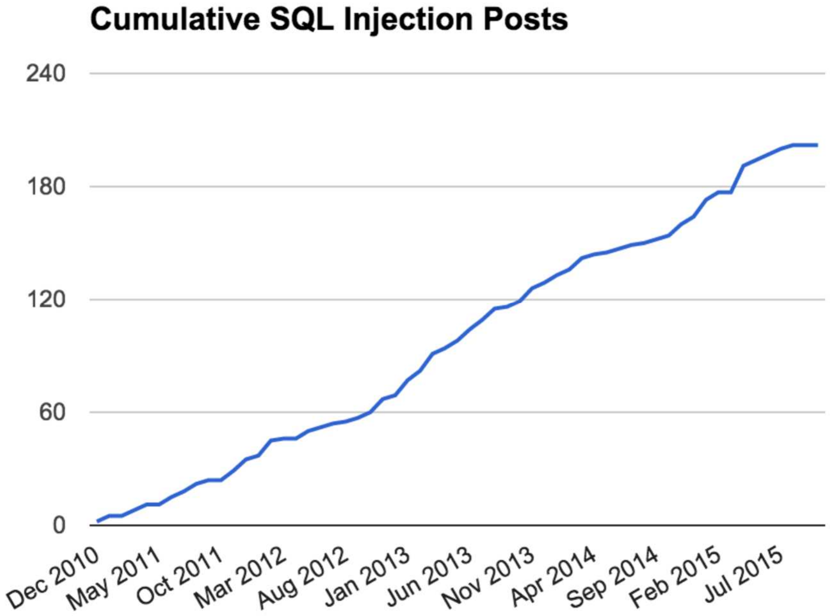


**Figure 11: SQL Injection Postings in Community B**

Of the 202 total postings related to SQL injection observed in Community B, five were found to be tutorials.  These tutorials were posted by change agents seeking adoption of the technique within Community B.  Two of these tutorials demonstrated a number of the innovation factors as members experimented with the technique, and are described in Table 13:

**Table 13: SQL Injection Innovation Factors in Community B**

| Date | Factor | Sample Quote |
|---|---|---|
| March 2011 | Relative Advantage | I personally use MySQL because its free and works well with Apache and whatnot. It also got a good syntax. It is also the most used engine so its what you will most likely encounter when doing injections. All SQL in this tutorial will be MySQL. |
| | Compatibility | Awesome tutorial, thanks!!!! I have to mess with this more, I was already doing all the preventive measures recommended in this tutorial but even when I take prevntive measures away I still can't break my code, I'm gonna have to do a lot more studying!!! example of some code I was trying to break (my dad owns a limousine company, I wrote this for him to keep track of maintenance on the vehicles) |
| | Complexity | well written [redacted] but the securing part take time to see it and if you are a newbie like me its hard to see all those codes |
| | | I must say that this is a HQ tutorial. I will expect nothing less from an admin. You made it very clear what you are saying and i understood it all. But i do have a lot of experience in SQLi so i dont know how it looks from a newbie. |
| | Trialability | I try this tutorial.. and have problem. First: when you try write http://[redacted]/index.php?id=17+ORDER+BY+5 or http://[redacted]/index.php?id=17+UNION+ALL+SELECT+1,2,3 you get nothing.. I solved this problem with ...?id=17' UNION ALL SELECT 1,2,3# but if you write in url in my case # dont works so I change it to %23. like ?id=17' UNION ALL SELECT 1,2,3%23. |
| | | Ok Thanks =] As we talked pm you said there is no CMS panels in that website i pmed you and said to check for XSS vuln. Well i didn't founded XSS vuln too :\ |
| | Observability | You mean the query union select and the sql server return all 3 columns. I may ask that why it output "2", "3" on the webpage rather than anything else, for example columns name. In some other tut, this query is said to find vunerable column, and u make difference. Sorry for my bad english |
| | | It outputs 1, 2 and/or 3 because that is the values you selected (SELECT 1,2,3). When you select a number instead of the column name, it takes the values as is, and prints them directly. You could also do SELECT 'a', 'b', 'c' and it would print 'a', 'b' and/or 'c'. You do this to see the changes in output, so that you know where to look and what is being printed in the next step: Actually retrieving data. |
| | Image | Videos are shit. You do not learn SQL injection without knowing SQL... this way you can learn few commands and that's it! Only if you know SQL you can make queries and inject them... ugh. |
| November 2012 | Relative Advantage | Advantage of using this script : I've given a textbox to enter the URLs. You can enter more than one URL to be checked. Very often you come across huge lists of URLs and people say that they are all Vulnerable to SQLi. Well, this is what you'll need then... |
| | Compatibility | The error message checking you are doing is a bit.. Broad. Many sites do contain the word "sql", "error" and "()" without actually having a bug/vulnerability. |
| | Complexity | RFI/LFI would be easy to do with your script, just replace the entire argument(s) with various LFI and RFI attack strings and look for the things mentioned above. For RFI you could try http://google.com and look for something google.com always has on their home page or something.. XSS would be as simple as replacing the ' with '"/><img src="herp.png" /> and look for "<img src="herp.png" />" in plain-text in the source. CSRF is a bit harder. And probably will not be very useful/practical/easy to make in this case. |

| | | |
|---|---|---|
| | Observability | Scanning only the last parameters for sqli is not a best way. May be the other parameters in the link is injectable too. Do you plan to fix it :)? |

Interestingly, the first tutorial in the above table was written by one of the administrators of the site. It was observed that a significant amount of praise was provided by more junior members as a response (such as stating that the tutorial was “HQ”, or high quality) and that the high status of the poster in the social system of the community appeared to encourage other members to experiment with the technique. As can be seen in the reference to “pm” (or private message) during the trialability phase, communication about this innovation occurred both over mass media public postings seen by all as well as interpersonal, private messages between the original poster (the change agent) and the potential adopter. As a comparison, the second tutorial was posted by a more junior member and the responses were more critical. The second tutorial involved this junior member who reinvented the innovation by incorporating SQL injection into his own custom script and is an example of enthusiastic adoption of the technique, even though others were critical of his implementation.

The introduction of the SQL injection technique to Community B appeared to have been in response to a pre-existing hacking need, as described in the tutorial posted by Community B’s administrator: “So, if you are able to set your own requirements in the query, we can also do an information retrieval injection, which in the end is what SQL injection is all about. Getting information you are not supposed to… This is because we asked for the ID which is 23, then the subject which is ‘admin passwords’ and then the text which is ‘abcabcabc’.” The goal of this tutorial was to teach members how to hack into vulnerable sites to obtain information to which they were not entitled – and this information was often the administrator password to the target website. This type of information is a frequent target of hackers and was a legitimate need before SQL injection was introduced into this community, as evidenced by the number of hacking-related postings observed prior to this date.

It was also noted in numerous other postings about SQL injection within Community B that the ability to engage in manual SQL injection versus using automated tools (such as Havij from Community A) affected the overall image of the member. Members who had the skills to conduct manual SQL injection attacks were seen as superior to members who utilized automated tools or who did not have complete knowledge of the principles behind the technique, as noted in the quotes below, obtained from a number of individual postings about SQL injection:

- *"Please, do me a favor, before you go ahead and follow tutorials on this tool. Read atleast [redacted] tutorial on sql injection. Do you want to be a TOTAL skid???, no!. So get some knowledge before accessing the tools of the trade. This is one of the most advanced opensource sqlinjection tools available. Tough you need to know some basics about sql injection and sql, sql servers, and server side coding to load this weapon."*
- *"I always tell them to learn to create before learning to destroy - because if you know how to create, you can think of more productive/effective ways destroy stuff and I like creating more, because then you have something to show. Like those skiddies doing SQL injection without having a clear picture of wtf SQL is - this just makes me sad, really."*
- *"Also in their arsenal, Anonymous uses extremely prevalent website vulnerabilities (from highest to lowest success: SQL injection, cross-site scripting,*

*file inclusion, directory traversal, general server misconfiguration, password/POST form cracking). Nothing of real interest or innovation. Most Anons only have an exceedingly basic, fallacious and surface knowledge of their actions and use readily configured popular security suites (Havij, sqlmap, Cain & Abel, Metasploit, Nessus, nmap, etc.)"*

- *"Oh, those are awesome! You only have to insert the link and hit a button, meaning that you don't even need to know any actual SQL Injection. I love those "H4x0rZ 1t n0wz" kinda programs. Why learn? That's wasting time, right?"*
- *"You can still find hundreds of easy SQL injections you just have to be crafty in how you.... craft your google queries. Also don't use NMap or Metasploit! Make your own versions so you truly understand what you are doing when you use those tools, once you've done that, then you'll no longer be considered a script kiddie."*

As in Community A, a number of postings about SQL injection appeared to transit the innovation-decision process as knowledge of the innovation spread and individual members evaluated the technique. An example of this process is described in Table 14, using the same initial tutorial from March 2011 as described in the previous table.

**Table 14: SQL Injection Adoption Phases in Community B**

| Date | Phase | Sample Quote |
|---|---|---|
| March 2011 | Knowledge | [initial post – tutorial] |
| | Persuasion | when I took away mysqli_real_escape_string, is_numeric, and the " for the var, I still couldn't inject :( |
| | | I appreciate the update, the information is understandable and simply well written. I have read and learnt some through this tutorial. |
| | Decision | Interesting article , Best way to master SQL injection is learning sql first and study whats happen behind the scene when we injecting . |
| | Implementation | euhh i'm a very bigenner in hacking lol, but i'm good at coding, so my problem is that i can't find tergets (i mean website where i can apply what you just teach as here) any ideas !!?? |
| | | I also hack with sql injection and somtime it show no column with union all select. I wonder it has not any text type column (a table always has text column, is that right?).Thank for answer |
| | Confirmation | Thanks for answer. Well it was the only user table on the database server (i pmed you link of that site) And password is not hashed. I Don't know where is problem... and that happens always when i get admin logins... i go to control panel and it says login invalid :\ i don't know what im doing wrong |
| | | Ive got an algorithm for this in C++ (console app) ask input->getstring = x -> search url : x + "sql injection stuff" -> if error output "vulnerable" -> then search for tables names and that stuff I think that I could make it :P |

In the innovation-decision process described above, the innovation is first introduced in the knowledge phase. In the persuasion phase, members form favorable or unfavorable decisions about the innovation – in this case, both types of opinions were formed as members attempted to use the technique. In the decision phase, the members decide to adopt or reject the innovation, and in the above example the member is shown endorsing the original poster’s methodology, which involves learning a significant amount about SQL before attempting to use it in attacks. In the implementation phase, members express their experiences when they use the innovation. Finally, in the confirmation phase, members declare their intention to continue to use the technique – in the first comment the member succeeds at obtaining the administrative username and password to the website being attacked, but is unsure of what to do next (thereby demonstrating the success of the technique), while in the second comment the user describes his intention to reinvent the technique by incorporating it into a C++ program he intends to write.

This initial tutorial by the community administrator was seen as a great success – indeed, for years afterward, when junior members would ask about the SQL injection technique, more senior members would direct them back to that same tutorial. As a result, it appears that Community B adopted the SQL injection technique.

#### 4.2.3.3 Community B – DDoS Attacks

The next innovation studied within Community B was the technique of distributed denial of service (DDoS) attacks. DDoS is a popular method of attack, where the attacker harnesses the power of multiple computers to send a flood of traffic to the intended victim (Li, 2004). These attacks frequently overcome organizations' IS security countermeasures by sending massive amounts of legitimate-seeming traffic, bypassing firewalls and other devices. The technique described in Community A of amplification attacks is a class of DDoS attack, and the booter tools described in that community utilize numerous types of DDoS methods to take down their targets. For Community B, the innovation of DDoS generally was examined, including related methods such as amplification attacks, reflection attacks, booters, and the popular tools "LOIC" and "HOIC" (Shiaeles & Papadaki, 2014).

As before, an analysis of the number of postings over time was conducted. The results of this comparison can be observed in Figure 12.

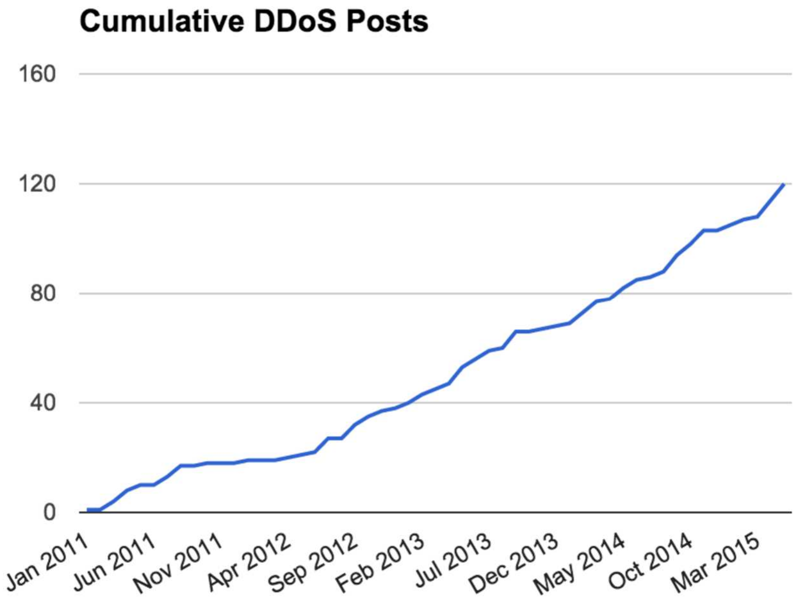


**Figure 12: DDoS Postings in Community B**

The diffusion of this innovation is close to linear, which is surprising given the low opinion of the technique expressed by members of Community B. The following quotes, taken from a single month of postings on the topic (July 2012), are representative of the general opinions expressed by members on the topic of DDoS:

- *"I can tell you right now you will not get any positive reactions from any respected member on this site when you spend your time ruining peoples work for no gain, just because they run software on their server that is vulnerable. I have a feeling you think DDoSing a site is pretty awesome too. I feel sorry for ya bro :-\"*
- *"You can't expect to be welcomed to a community where you think hacking includes defacements (and possibly DDoS however you never responded to that) Defacements take no real skill at all which is why a bunch of 12 yr olds are doing it (along with DDoS')"*

- *"Writing about DDoS is not very interesting unless it is something new, and even then it is not really all that interesting."*

There was a significant theme of public shaming for those members who professed to conduct DDoS attacks, and the image of those members was significantly reduced. Because of this, there were not any conversations regarding the diffusion of DDoS innovations where members experimented with and adopted or rejected tools or techniques. Likewise, a pre-existing hacking need related to DDoS appeared to be non-existent. Instead, when members attempted to seek help with DDoS attacks or solicit other members to engage in them, the postings were moved to a section of Community B called the "Board of Shame" for public ridicule. Member B2 stated about this section of the community, "Public shaming is one of the most used tools on [redacted] in dealing with members who ask 'stupid questions'. If the thread or the responses are funny enough, the thread may even be moved into the Board of Shame, which is a public shaming tool."

There were some discussions about innovative techniques, such as reflection or amplification attacks, but these discussions centered on detection and defense, or learning about the technical principles behind such attacks, and not about the adoption of the technique itself. Member B1 stated that these discussions were allowed because while Community B members "hate ddos as a service, it's not frowned upon talking about techniques, tools and use-cases of ddos if it's related to learning and sharing some knowledge".

Generally, those members who claimed to engage in DDoS attacks were labeled as "script kiddies" and shunned, and there existed a strong social pressure to not speak about such attacks, with common quotations such as “I would never DDoS or anything like that, that's for skids” and “Just looks like a shitty DoS tool (I believe people call this particular type a ‘shell booter’). Not even written well at that”. Member B1 echoed this sentiment, stating that “even though hacking discussions are what should be around here, we don't condone illegal activities and most of us hate DDoS as it's just stupid.” Member B2 added that DDoS is “seen as something that skids do/buy. Requires no skill and is lame.”

Because of this behavior, it appears that Community B as a whole has rejected DDoS innovations. This type of rejection is a passive rejection, where the community has rejected the innovation outright, without considering it for adoption (Rogers, 2003). It is possible that individual members may adopt the technique in secret, but Community B was observed as a place where innovation into DDoS attacks did not occur. This in itself is a novel finding as the community rejected an entire class of hacking techniques based upon primarily the image factor, although complexity also played a role as DDoS attacks were seen as too simplistic to be interesting.

#### 4.2.3.4 Community B – Crypting

The third innovation studied within Community B was the technique of crypting (or in tool form, crypters). Crypting is the process of encrypting malware executable files so they are not detected by IS security countermeasures such as antivirus programs and is a parallel process to packing, or compressing, malware, which is also frequently done for obfuscation (Yan, Zhang, & Ansari, 2008). Between 80 to 90 percent of malware samples are obfuscated in such ways in an attempt to deliver malicious payloads to targets before they are detected (Lyda & Hamrock, 2007). Using the slang of these hacker communities, the goal of the malware is to make it “full undetectable” (or “FUD”). Because crypting and packing techniques are often closely linked, searches for threads describing both will be conducted. The processes of crypting and packing, and of creating crypter and packer tools, are quite popular within hacking communities and were observed extensively within Community A, although less so within Community B. Figure 13, below, describes the prevalence of discussions regarding crypting and packing techniques within this community.

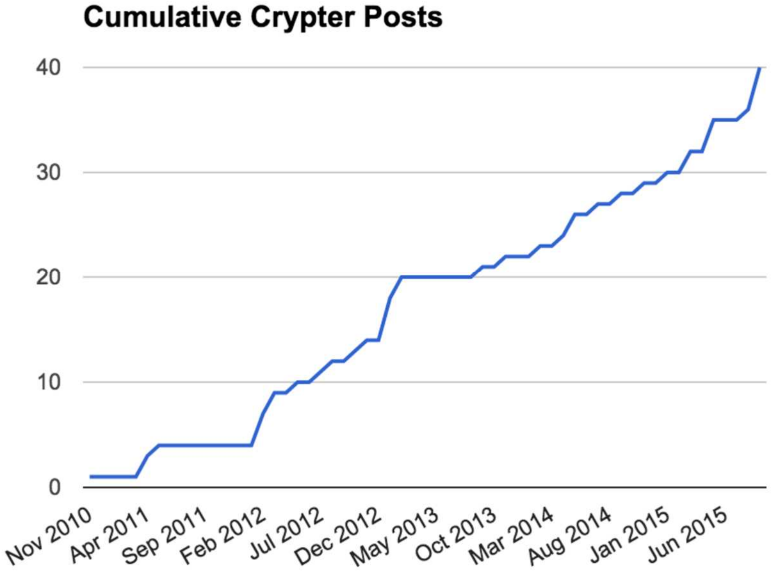


**Figure 13: Crypter Postings in Community B**

A review of innovation diffusion activity surrounding crypter-related threads was illustrative. Of the 40 total conversations related to malware crypters, four were identified to be tutorials. Two of these threads were selected below in Table 15 as examples where innovation factors were clearly discussed.  Both of these threads involved members who were releasing crypter tools that they had developed to the larger community.  It was interesting to note that posts about crypters or packers that appeared to have been authored by "script kiddies" that were seeking help with obfuscating clearly malicious software, such as Trojans, were moved to the "Board of Shame" similar to what had been observed in the DDoS section above.  Posts about these innovations that were technically novel or were geared toward the sharing of knowledge were commented upon, and perhaps even experimented with or improvements suggested, even if the ultimate goal was likely for malicious purposes.  This may indicate that image plays a role in the adoption of various techniques or tools; however, the image in the case of Community B, that of technical superiority or mastery, is different from the image observed in Community A, that of "street cred" within the community.

**Table 15: Crypter Innovation Factors in Community B**

| Date | Factor | Sample Quote |
|---|---|---|
| May 2015 | Relative Advantage | Being a noob looking for stub crypter sources that I would finally be able to understand, I stumbled over this (rather imperfect, but tiny & lucid) piece of code… I rewrote most of the mess and thereby improved it quite a bit, I guess. Now I'm happily adding this project to public domain… Nevertheless, with some initiative of your own (i.e. adding/changing about 20-30 lines of code), you might be able to use this code as a straight basis for your very own crypter, which then actually IS fud! |
| | Compatibility | Nothing is obfuscated or protected at runtime. The code is plain visible in memory while being executed. It must be, otherwise it could not be executed. The only thing that comes close is a protection mechanism that only deobfuscates small parts of the code at a time during execution. But even then the small part is still plain. These "scantime protected" files are actually nothing more than file droppers created by a builder. Or if you want terms that are not associated with malware, then they are wrapped files. In your case the program that wraps the files is a bundler, because it will turn two executables into one executable. These terms scantime and runtime crypter are misnomers and I still wonder why everyone just swallows them without complaining. |
| | Complexity | Just to clarify for everyone: You wrote/modified a crypter/binder that appends two executables in an encrypted form to the overlay of the stub. The stub decrypts the overlay, writes both executables as temporary files to disk and executes them. Correct so far? The PE format is one of the most complex formats that I know off. So I am not sure what you mean if you say it is easy to f*ck up and therefore not complex. I think it is the opposite. You can f*ck it up so easily because it is complex and you will make mistakes easily while trying to modify a PE. |
| November 2010 | Relative Advantage | I made this when I was bored this summer. It cyphers files by adding or xoring one byte (user-selected). It includes a stub for working with exe's. |
| | Compatibility | And that's why it perfectly fitted my needs :D i.e. abilities. Many other stub sources I've found were either .net crap or just way above my comprehension skill. (But since you helped me understanding general stub concepts, they're not so hard to analyze anymore. Thus I've already got sum new projects in the pipeline ;) ) |
| | | Note: Its a scantime crypter, not runtime. But still a good example :) A scantime crypter will only "protect/crypt" your file when the file is not running. Once you run it, most AV's will detect the temporarily created file. A runtime crypter will make the file undetectable at all times. |
| | Complexity | It's not that hard to modify to make it a runtime crypter. |
| | Trialability | Here's [name] crypt v2.0, written by me: [link] I hope you don't feel copy&pasted or anything :P and that it was okay to re-use your name |
| | Observability | However, after a few minor improvements, this script can do surprisingly well! Just got a 1/33 on nodistribute with some generic keylogger, hihi |

In the first example from Table 15, originally posted by the member in May 2015, only the factors of relative advantage, compatibility, and complexity were observed in the discussions. This may be because the innovation was initially posted by a junior member, and because the responses of more senior members were not encouraging and appeared critical and dismissive of the innovation. As a result, other members were uninterested in pursuing adoption of the tool. The hacking need that this innovation fulfills is clearly described, though – a desire to protect malware at scantime and at runtime from antivirus software, to allow the delivery of the malware to the target. Such a hacking need predated the introduction of this technique into this community.

In the second example, from November 2010, a senior member of the "VIP" level posted the initial innovation, and other senior members were more enthusiastic about the innovation and it was eventually adopted. This may highlight the importance of senior members, both as change agents when championing innovations and as gatekeepers through which other innovations must pass, and where a negative review or comment could result in an innovation not being adopted. This example is expanded upon in Table 16, as Community B brought this tool through the innovation-decision process before adoption.

**Table 16: Crypter Adoption Phases in Community B**

| **Date** | **Phase** | **Sample Quote** |
|---|---|---|
| November 2010 | Knowledge | [initial post] |
| | Persuasion | Great for people looking for open source crypter. Thanks for sharing. |
| | Decision | This is awesome bro. I will for sure use this as an example |
| | Implementation | What are you working on. I was planning to create something like this for my encoder/decoder. |
| | Confirmation | Here's [name] crypt v2.0, written by me: [link] I hope you don't feel copy&pasted or anything :P and that it was okay to re-use your name |

The innovation described in Tables 15 and 16 was a “Simple Crypter + Stub” written in the C programming language, according to the innovator who posted the code. Complimentary postings were made by other members in the persuasion phase, and intention to use was stated by a member in the decision phase. In the implementation phase, another senior member stated that he planned to develop a similar implementation in his own tool, and in the confirmation phase, another member reinvented the code by adding new features and releasing the improved version to the community. This is an example of a crypter innovation that has been adopted by Community B.

Member B1 confirmed these observations about the influence of senior members, stating that “but with power comes responsibility, so maybe that's why we are respected a bit more”, and “if more of respected members support the idea it might get developed more”. Member B2 did, however, argue that ideas with merit would be respected in any event, and related “if you're a newfag or an oldfag, if you're idea is brilliant then it's supported, if it's stupid then you are told that it's stupid”, but that the status of the poster may influence how other members react, stating that the “reply wording may vary on the OP status, i.e. I wouldn't get told to fuck off if I posted some stupid suggestion in a serious context”.

Member B2 provided additional details about crypters, stating that he had authored one of Community B’s most respected papers on crypters. In this paper, Member B2 claimed that good malware writers should not need to use crypters, as if the author is skilled enough the malware will be undetectable without the need for an outside crypting program. Member B2 stated, “Since that paper I saw others posting that opinion whenever the topic came up. It was not like that before. I believe before that you could publicly show off your own crypter, but you could not ask that other people create you a crypter for free without getting shamed." This points to the influence of senior members upon the adoption process – Member B2's statements on the image of crypters led to a rejection of the technology as a whole. Member B2 also stated that the terminology used by members made a difference regarding the diffusion and adoption of a particular innovation, stating "if a new member asks something about a packer it is probably not as bad as asking for a crypter, although crypters are a subset of packers.

Everything that is associated with being typical for [Community A] is despised. That is a certain language (FUD, Scantime Crypter, etc) as well as certain types of tools and questions." Therefore, it seems that *how* an innovation is communicated within Community B appears to be important, not just the type of innovation itself. It appears that in general crypters as a technique have been rejected by Community B, although some individual crypting tools are eventually adopted. This was an active rejection as per Rogers (2003), as the community did experiment with the innovation before rejection.

#### 4.2.3.5 Community B – Bitcoin

The final innovation studied within Community B was Bitcoin. Bitcoin is a virtual currency that is decentralized and anonymous, and utilizes cryptography to both secure transactions and to generate new Bitcoins through a process referred to as "mining", and as such is referred to as a "cryptocurrency" (Huang, 2013). This is a fairly broad innovation with many applications, and to focus more upon areas of relevancy to this research only malicious uses of Bitcoin – such as malware that steals Bitcoins, or botnets that use compromised computers to mine Bitcoins – will be examined within Community

B. Note that to the end goal of these techniques is to obtain Bitcoin, a valuable resource, and the systems which contain Bitcoin are often protected by IS security countermeasures themselves. Bitcoin mining through malware is extremely popular and Huang et al. (2014) estimated that 4,500 Bitcoins, with a street value of nearly $4.5 million (depending upon when the mining occurred) had been mined using botnets. Bitcoin is also subject to other types of criminal activity, including money laundering, intentional market disruptions, double-spending of the same Bitcoin, and other attacks (Huang, 2013).

A quantitative review of the number of postings within Community B related to malicious use of Bitcoin was conducted, and the results can be found in Figure 14, below.

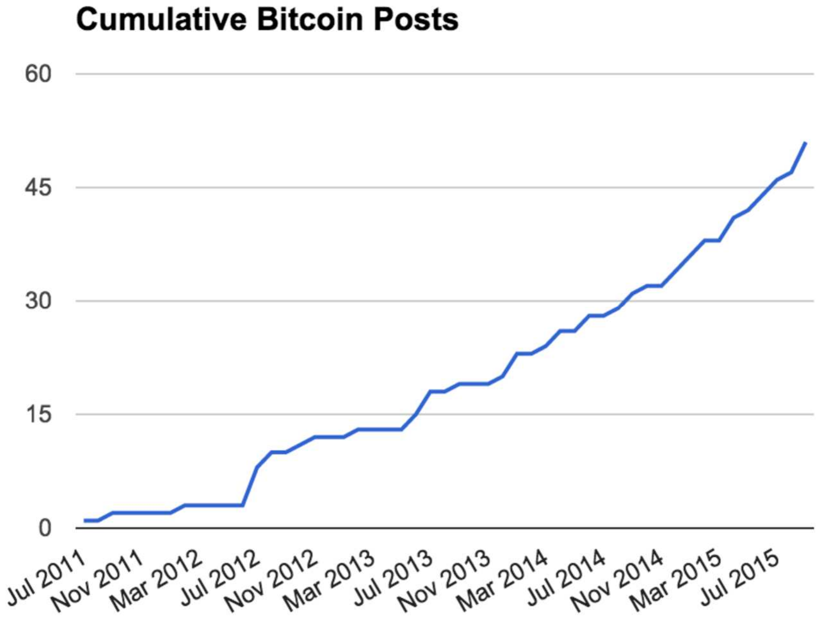


**Figure 14: Bitcoin Postings in Community B**

Many of the postings observed in Community B related to Bitcoin mining using botnets or purchasing hacking-related services, such as VPN connections or proxy servers, with Bitcoin. One thread was observed from 2012 pertained to Bitcoin mining – this was a thread that emerged relatively early in Community B's discussions about this innovation and many members were still experimenting with the technology. This thread was a demonstration of both many of the innovation diffusion factors as well as the innovation-decision process as these members decided whether or not to adopt the innovation, and can be observed in Tables 17 and 18, below.

**Table 17: Bitcoin Innovation Factors in Community B**

| **Date** | **Factor** | **Sample Quote** |
|---|---|---|
| August 2012 | Relative Advantage | Bitcoins are a digital currency that you can buy stuff online with. They are going for about 1 bitcoin for 9 USD at the moment, and still rising. |
| | | It is a cool concept, no governments fucking shit up and so on. However, the big players are winning on this currency as well. |
| | Compatibility | So doing bitcoin mining on your own comp.just ain't worth the time tbh, however joining a bitcoin mining pool could be worth it. you'll have to check that out yourself. If I did a poor job explaining thia, ask for clarification and ill do what I can. Cahse I do think the hows and whys of bitcoins are quite confusing lol |
| | Complexity | Bitcoins are a currency that is transferred virtually anonymously through complex mathematical algorithms called blocks. These blocks are super complex, and must be solved in order to allow future transactions. Bitcoin mining is the act of using your GPU to solve these equations, for which you are rewarded 50 bitcoins. This is also how bitcoins are added to make the available currency pool larger. These bitcoin.generations are severly reatricted however due to a number of factors. |
| | | Given the current rate of ASICs coming up on the network, GPU mining of BitCoins has become retardedly hard. I barely get shit done with my 7970, and a friend of mine running CF'd 7970's is also saying it's starting to become too slow. |
| | Trialability | ok how does a mining pool work , can i make my own with a bot-net ? |
| | Observability | and what's the point in investing in bitcoin when there is stock and real estate? 10 % a month is not difficult in stocks. |
| | | Very true. But bitcoins can also be invested in without any cash. Just start mining, and either try to do it yourself or join a mining pool. Personally I don't think it's worth it atm, but other people seem to think it is. |

Interestingly, there was little discussion about the image factor in any of the Bitcoin discussions, even though image played a significant role in the other innovations studied for Community B. Adoption, or lack thereof, did not appear to affect the social standing of a member. The decision process for this same thread is described in Table 18:

**Table 18: Bitcoin Adoption Phases in Community B**

| Date | Phase | Sample Quote |
|---|---|---|
| August 2012 | Knowledge | [initial post] |
| | Persuasion | While investing now won't be quite as good as it was when bitcoins first started, it could still be worth the cash as bitcoins rise in value. Pay $6 each now and sell for $20 each in 5 years, something along those lines |
| | | bitcoins... when paypal ain't enough. really tho, the idea is bullshit. I don't understand why people are so freaky about it. |
| | Decision | If someone is still intrested in mining, look for mining hardware like the ones from butterflylabs.com I have started to think about buying these some weeks ago and there are really good feedbacks. |
| | | My idea was to start with a 270$ miner and after 1 month or 2 buy one again, and some months later guy the bigger one and so on. |
| | Implementation | My brother got into bitcoins while the where like $0.10 and he, thinking it wasnt going anywhere, got like 50 bitcoins. Then they shot up to like 20-30 a piece. He was really pissed because he could have made a fortune. |
| | Confirmation | I had bad experiences with BitCoins. That stuff is unstable man, wouldn't waste my time on it. Not yet anyway. I used to do bt mining with some of the most powerful GPU in the world and still it was a waste of time. |
| | | Actually out of the mass of altcoins that have cropped up, by process of elimination, me and my friends ended up only with LTC as the most viable replacement for the shitty system that BTC turned out to be. |

Even though the feedback in the confirmation phase was not positive, the innovation appears to have been adopted by Community B eventually, as numerous discussions about the use of the topic were observed in the years following the above posting. Some of the same members who made the contributions to the original thread ended up being enthusiastic proponents of the innovation. The need for this innovation – at least, for the cybercrime-related aspects of Bitcoin – is less clear. It seems that because cryptocurrency was a new invention, there was not a pre-existing need for malicious activities around it, or mining Bitcoins; however, there certainly was a pre-existing need to make money from online activities. The ability to make money from resources such as botnets was certainly a need that existed among members of Community C, and therefore this innovation appears to have been introduced as a response to that need.

### 4.2.3.6 Community B – Summary

Community B was a significantly different hacking community from Community A, both in terms of overall attitudes of participants as well as innovative behavior. Image as a critical innovation factor was found to be important within Community B, as was observed within Community A; however, the type of image was different. Within Community A, adopters did not want to be perceived as "skids", but discussion of adoptions that fell within the domain of script kiddies (such as Havij) was still robust. Within Community B, on the other hand, script kiddies were also panned but the community as a whole tended to shut down posts that outwardly malicious with no technological merit. Community B appeared to be much more focused upon *learning* and upon discussing novel technical hacking innovations, whereas Community A appeared to be focused upon the executing the operations of attacks and upon vendors selling services – some learning did occur in Community A, but there was a significant amount of attention paid to pure operations.

Unlike Community A, Community B did not have a financial aspect, and no vendors were observed selling tools or services to other members. This meant that an entire class of innovators – that is, vendors who sought adoption of their services – was absent. According to Member B2, this was by design – "it is on purpose that we don't have a market place here as it attracts the wrong kind of people. Those, who need to buy 'hacking services' are no hackers." To discourage these types of members, Member B2 stated that threads about selling tools and services, or about "RAT/crypter/bot-builder/other skid shit tutorial here" were routinely deleted by moderators, or moved to the "Board of Shame". Exceptions to this rule are made when the tools being described are actually written by the member, and if the discussion is very high quality. It was also observed that members of Community B had an extreme dislike for the members of Community A, deriding them as “skids” and hackers with little skill. In general it appeared that Community B was a medium-high skill level community, as its focus upon education seemed to result in a lack of active, highly-skilled attackers that will be discussed in Community C, while still being above the attackers observed in Community A.

In general, innovations were introduced in response to pre-existing hacking needs. An exception to this rule was DDoS attacks; this innovation was strongly discouraged and as such the members of Community B had no existing need for it. Communications channels utilized within Community B appeared to be those developed by the community itself. Unlike Community A, Community B only utilized forum postings, private messages, and a community-run Internet Relay Chat (IRC) channel to engage in discussions. Outside services, such as vendor sites or third-party chat networks, did not appear to be utilized to a great extent. In addition to the channels of communication, the language used in the communications was also important – for example, posts on crypters that utilized slang commonly found in Community A were blocked, but posts that did not use this slang were allowed to continue.

None of the innovations examined were developed (from a software engineering standpoint) within Community B – like Community A, the innovations were introduced and discussed within the community by members who may have sought wider adoption, but in the case of Community B had no vested interest, aside from perhaps seeking elevation of image by discussing and championing a popular innovation. Senior members also played a significant positive role as change agents in the adoption of the studied innovations – both if senior members originally proposed the innovations, and if senior members later provided endorsements for the innovations. A senior member contributing to a specific thread often made the difference between a positive reception culminating in adoption or a negative one culminating in rejection. As before, voluntariness did not play a role in Community B's adoption of thee innovations.

### *4.2.4 Case Study Analysis – Community C*

#### 4.2.4.1 Community C – Overview

The third hacking community studied, Community C, was created in 2005 and boasts over 35,000 user accounts. Of these only approximately 6,800 were used to post messages to the community – while similar in overall user accounts to Community B, Community C appears to have twice as many active members. The stated purpose of Community C was overwhelmingly on malware, and specifically upon programming and developing malware, with focus upon code-level technical details. Community C also had a hierarchy of members, with "Junior Members" at the bottom, and "Senior Members", "Moderators", and "Admins" at the top. Once a user has registered for an account, they have access to the entire community.

Based upon the results of the expert panel and the findings from the previous two communities, the following innovations were studied in Community C:

- SQL injection
- Bulletproof hosting
- DDoS attacks

- Crypters

Data collection and analysis for Community C was similar to the methodology employed in Communities A and B. As before, interviews were conducted with two senior community members. These members were identified following a review of forum postings related to the innovations under study. Note that unlike in Communities A and B, approximately ten members were contacted over private message by the researcher before two agreed to participate – some contacted members refused to participate, while others simply did not respond to requests. Member C1 is a moderator who joined in 2006 and has over 1,600 postings to the community. Member C2 is another senior member who joined in 2007 and has over 2,000 postings. Both of these members remain active within the community and have experience with the innovations being examined.

### 4.2.4.2 Community C – SQL Injection

The technique of SQL injection, described in detail above in relation to Communities A and B, was the first class of innovation reviewed for Community C. As a hacking community, Community C did contain postings about specific tools and techniques related to SQL injection, although fewer than in the previous communities due to Community C's general focus upon malware. Figure 15 shows the analysis of these postings, of which 28 were observed.

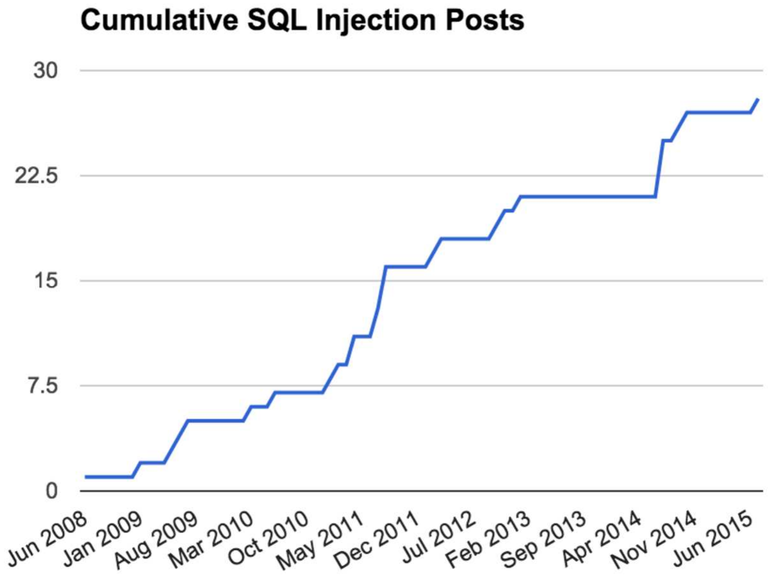


**Figure 15: SQL Injection Postings in Community C**

Similar to the previous communities, members were derisive of others who utilized automated tools such as Havij, but to a much lower level than what had been previously observed. Numerous members stated that using "your brain" is the best approach, but even then would allow for tools: "Find the SQLi with your brain and then feed it to sqlmap for faster exploitation." As was observed in Community B, a significant amount of vitriol for Community A was noted, with members of Community C expressing disdain for unoriginal or unsophisticated postings and telling the posters to "go back to [Community A]".

While Community C had fewer discussions regarding SQL injection innovations than was observed in the prior communities, there were still a few threads where innovation diffusion factors were observable. As an example, Table 19 contains a description of the innovation factors observed in a discussion regarding a SQL injection vulnerability that one member had found in a popular banking Trojan botnet known as SpyEye, which allowed him to take control of the botnet from the original owner:

**Table 19: SQL Injection Innovation Factors in Community C**

| Date | Factor | Sample Quote |
|---|---|---|
| April 2011 | Relative Advantage | Exploit ./sqlmap.py -u"http://x.x.x.x/formgrab/frm_findrep_sub2.php?id=1" --file-read=/var/www/formgrab/config.php define('DB_USER', 'root'); define('DB_PASSWORD', '1234TTab123xsdesd568'); # Admin define('ADMIN_PASSWORD', 'hdjsd898oiSDKfsdf32423ncc'); |
| | | frm_findrep_sub2.php?id=1 this file is sql injectable. |
| | Compatibility | Ok guys, does the latest panel with build 1.2.99 from RED work? I guess ill have to try..I wont forget to check that SQLi function and patch it =P |
| | Complexity | I didn't found any other remote-exploitable SQLi and btw if gribodemon it's an idiot why you publish this?by doing this you help gribodemon improving spyeye security.... |
| | | Gribo very noob coder. Found this months ago. |
| | Observability | why are you an fucking idiot an publish this?i found this two months ago and kept it for me,few friends for not getting patched,now you did it....btw in the main/form panels are also some persistent/non-persistent XSS but you can't exploit them remotly… |
| | | thanks for this share. Thx to this i catch one of the big gribo friend botnet 15-33k online. |
| | Image | thanks for sharing. About [redacted], he just wanted to show us that he knew about this exploit and this post is not a surprise for him. Wanabefamouskiddo |

The original poster describes the SQL injection vulnerability and shows that the database and admin passwords for the botnet are obtainable, demonstrating the *relative advantage* of the technique. This advantage also demonstrates a similar hacking need – obtaining the administrator password – that had been observed in Communities A and B in relation to similar techniques. Such a need was pre-existing in this community as members had desired to commit intrusions via administrator accounts prior to the introduction of SQL injections. A member asks about the *compatibility* of the technique with another version of the botnet that he is attempting to target, and other members discuss the relative lack of *complexity* in the vulnerability, while also adding that no other similar vulnerabilities were found. For *observability*, one member describes the results of his using the technique to find and take over a large SpyEye botnet, while another member complains that posting this method in such a public fashion will allow the botnet creator to fix the vulnerability. Finally, a participant states that the member who complained was worried about his *image* because he did not want his credibility undermined by appearing to be surprised about the vulnerability. Note that *trialability* was not expressly observed but components of the experimentation of this technique were seen in the above postings, implying the members did experiment with the innovation prior to total adoption.

Other threads demonstrated the diffusion of SQL injections through the innovation-decision process, as described in Table 20:

**Table 20: SQL Injection Adoption Phases in Community C**

| Date | Phase | Sample Quote |
|---|---|---|
| January 2011 | Knowledge | [initial post] |
| | Persuasion | ps :in mysql user acc lvl right is important to run this comand sometime you can just read  and in mssql there is no limit ! |
| | | you wount hack any site with this program |
| | Decision | So basically you're telling me I can't do anything with it (apart from reading database entries) after I found an injection? -.- |
| | | Or get the administrator password, log in to panel, upload shell and root it. Also You can try mysql load_file to get config.php , get mysql password, connect to it and own. |
| | Implementation | Today, I wanted to modify a row and I got the following problem: According to the documentation of mysql_query, PHP only executes the first command in the query string, so I can't just append my second query to the first (with a semicolon in between ) |
| | Confirmation | @[redacted]: That's great, thanks! I'll take a deeper look into it later. @[redacted]: It seems I can't read any files at all (well, with the particular server I'm working on at the moment ). I don't know the DocumentRoot, though, so it could be that I just used the wrong paths every time... |

In the above conversation, a member is considering adopting the SQL injection technique but is seeking help from other members. In the initial posting, the member enters the *knowledge* phase by making others aware of his desire to learn about the technique. In the *persuasion* phase, other users attempt to persuade the member to adopt or reject the innovation – one member highlights the effectiveness of the technique within MYSQL and MSSQL, but another member dismisses the approach entirely. In the *decision* phase, the potential adopter was observed weighing the decision to adopt, but appeared discouraged because he can only use it to read information. Another member chimed in with additional approaches to take. In the *implementation* phase, the potential adopter describes that he is still having problems implementing SQL injection attacks, and in the *confirmation* phase, the adopter states that he will continue to try to improve his SQL injection technique. It is ultimately unclear if the adopter was able to resolve his issues with the technique and fully adopt, although this example conversation details the assistance that other members of Community C can provide when a member is deciding whether or not to adopt.

SQL injections as a whole within Community C appeared to have a lackluster reception, and it appears to be rejected as a whole by the community, outside of certain niche areas such as employing SQL injections to hijack botnets. This appears to have been an active rejection, as per Rogers (2003), as some members did experiment somewhat with this method before rejection. This determination was made based on the small number of postings regarding the topic and the relative lack of innovation-related activity related to SQL injection attacks. Member C1 agreed, stating that “SQL injections aren't really that popular (here)”. This rejection of this technique appears to be because SQL injection is not compatible with the malware-writing needs of this community.

4.2.4.3 Community C – Bulletproof Hosting

The second innovation studied within Community C was bulletproof hosting, as was previously described in Community A. Bulletproof hosting is a necessary part of the distribution and operation of botnets, Trojans, and other types of malware (Kamluk, 2009) and would seem to be a popular innovation within Community C, given the community's primary focus upon malware. A review of conversations within the community resulted in a surprisingly low number of postings, however, which are reflected in Figure 16. This may be because the focus of Community C is upon the *authoring* of malware and not upon the *operation* of malware, and bulletproof hosting is certainly a component of the operation of malware infrastructure. Member C1 stated that many bulletproof hosting providers ended up being scams or simply were not truly bulletproof and the supposedly untouchable servers would go offline.

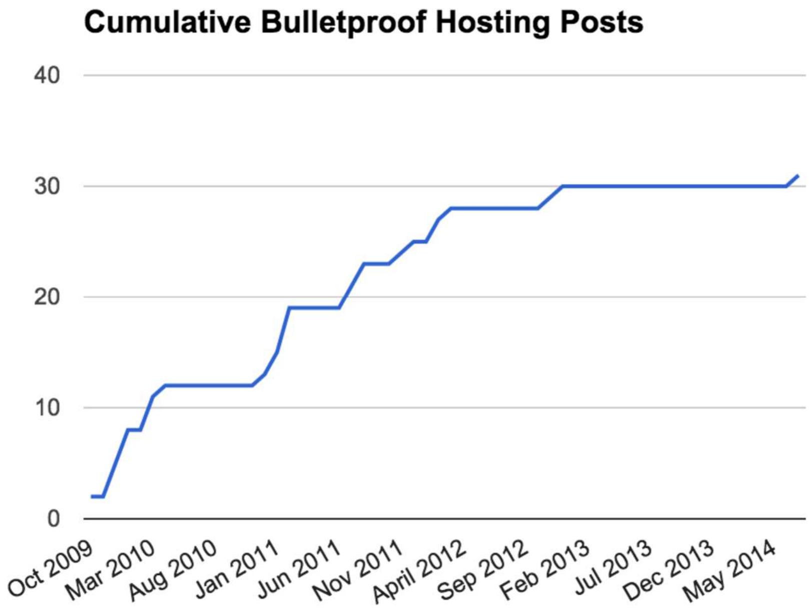


**Figure 16: Bulletproof Hosting Postings in Community C**

A total of 31 postings were observed, starting in October 2009 and ending in July 2014. No postings have been seen regarding bulletproof hosting since, which may mean that the members of Community C have either fully adopted the technique or abandoned it – or that it is less relevant to the community than had been thought. Member C1 stated that he believes this is because Community C no longer offers a dedicated marketplace for vendors, and as a result those offering bulletproof hosting services have migrated to other forums. Most of the 31 postings observed dealt with botnet hosting, and specifically the Zeus botnet, a banking Trojan similar to SpyEye. The hacking need for this innovation was firmly in place prior to the introduction of this technique – as the entire point of the community is around malware, one of the main problems as observed during the review of Community C was finding a suitable place to host it so victims can be infected. Member C1 stated that the innovation "was a necessity when web-based bots came out, when 'building' a bot you have to include a domain for the bot to connect to, if said domain ever goes down all the bots that were connected to it are effectively lost". Unlike Community B, the selling of services are allowed in Community C, and some of the historical postings related to bulletproof hosting were from vendors selling their services.

Within the bulletproof hosting arena, there were a few legitimate innovations observed within Community C. The first innovation involves a proposed method to make a normal webhost bulletproof, while the second innovation involves a proposed technique to host botnet command and control servers within the Tor network. Both are improvements upon the original bulletproof hosting idea, mostly because finding true bulletproof hosting is difficult, as explained by Member C1, and expensive, and the individual innovation factors are detailed in Table 21:

**Table 21: Bulletproof Hosting Innovation Factors in Community C**

| **Date** | **Factor** | **Sample Quote** |
|---|---|---|
| February 2011 | Relative Advantage | Well, I regularly set up HTTP controlled botnets on free webhosts, and after a few of my C&C panels got shut down and accounts suspended... I began to have a think… I would install my webpanel in a sub-dir made accessable, and then proceed to create a shitty website (generic_lolcat_website OR generic_bullshit_blog OR generic_vanity_site) of some form to 'cover up' or 'mask' the presence of my C&C… Just an idea for those on free hosting... Next I plan to do the same with Java Drive By/exploit kits... and host server.exe on the site... |
| | Compatibility | not everyone has a gazilion dollars to spend off on bulletproof hosting in foreign countries. |
| | | Like I said, not for serious botnets. useful for experimenting with new leaks on maybe 50-100 bots. In total I now have close to 1k live bots across almost 20 different 'panels', been thinking to get a stable bot + offshore BP and migrate sometime soon |
| | Complexity | or hacked one. it will last for a month before the owner or the sysad to find out.. |
| | Trialability | get cheap offshore plus BP domain which is pretty cheap also, if your host gets shut, just change nameservers on domain for new host so you don't have to spread again |
| | | im like the OP and not bothered to pay for hosting, the simple way is to have your bot connect to many free hosting sites to distribute the bandwidth, its no harm at all if you keep it legit looking. the best way is to just have LOTS of free hosting accounts and a checker to be able to tell when they get banned. |
| | Observability | I only use freehosts because I am not making money or anything from my botnets, they are moreso just for loading twitter-controlled-DDoS tools. When I get more serious and practiced I will be paying for offshore bulletproof hosts, but for now I am stuck with damn free ones... |
| | Image | Bullshit!!!!! You can't get reliable bp hosting for €5, you are obviously from [Community A] and believe the advertisements. Do you believe FUD for life too because its java? |
| August 2011 | Relative Advantage | Did you ever built a botnet with thousands of bots and was it taken down in minutes by the feds revoking your domain or nullrouting your servers? Do you wish this would never happen again? |
| | Compatibility | TORifier works with any bot, HTTP aswell as IRC. And RAT trojans too! You only need to be able to set the hostname and port on your own. TORifier was successfully tested with ZeuS, SpyEye, rxbot and poison-ivy. Except a higher latency (~1000ms) there were no downturns. |
| | Complexity | With TORifier you can easily upgrade your existing bot executable with a TOR tunnel, tunneling all the traffic trough multiple TOR-relays to your server, which is behind a hidden service. Only the onion-domain (a pseudodomain used for communication with hidden services) is known and noone can assign a real IP to this domain! |
| | Trialability | Price will be negotiated over ICQ, but expect to pay as much as you pay for SpyEye+Plugins. Remember, that this investion will lower your risk running a botnet dramaticly! |
| | Observability | you seem not to understand what a hidden service is, I added .pdf with |

| | | |
|---|---|---|
| | | setup instruction. Collector daemon works over hidden service, I tested it myself. Reverse connection features like ftp and socks5 will of course work outside hidden service, because they require speed. You should really watch the videos as you talk crap. |
| | Image | Lol TORifier is a windows tool. Windows's security is crap making it easy for anyone pwn your botnet. |

The first innovation received a lukewarm reaction within Community C, possibly because the technique was designed to be "not for serious botnets" – instead, the technique was designed for botmasters just starting out who don't have the money to pay for true bulletproof hosting. Some members defended the technique while others pointed to the availability of cheap offshore hosting that, while not bulletproof, might suffice. The image factor was not a huge concern, except when one member stated that bulletproof hosting can be found cheaply, and another member retorted that perhaps be belonged in Community A. As seen in Community B, the name of Community A is used almost as an epithet and is associated with a severe reduction in image. The second innovation was received more enthusiastically, due to the compatibility with the existing needs of Community C – developing malware for Windows-based victims – and because the complexity of the technique allows for better obfuscation of the botnet traffic, which means that it may stay undetected for longer. In this case, increased complexity of the innovation is a positive benefit.

A review of innovations that transited the phases of the innovation-decision process was also useful. Two innovations are described in Table 22 – first is another Tor-based malware communications service, and second is a traditional bulletproof hosting vendor who is advertising his services.

**Table 22: Bulletproof Hosting Adoption Phases in Community C**

| Date | Phase | Sample Quote |
|---|---|---|
| March 2011 | Knowledge | [initial post] |
| | Persuasion | excellent tutorial bro i must to install it soon |
| | | tor is not good to hide you work , they keep logs and FBI can demand the TOR owners to release any information needed to trace the identity of the user.It has happened before |
| | Decision | I don't understand, how do we go about the connection? :boss: do we install Tor and use our PC to host zeus and then put what Tor URL to be able to access it???? |
| | | I'd love to see this working with a botnet that is bigger than 100 bots. |
| | Implementation | I have a feeling that this is how I will be making my next botnet for tests, very interesting tutorial! Never thought about using TOR hidden services desite being more than familiar with it uses... |
| February 2011 | Knowledge | [initial post] |
| | Persuasion | ok nice service, offshore / bp, but bestiality, rape and child porn are a big NO on [Community C], so I'd suggest not advertising that. |
| | | There is no real bp service. So this means your full of shit man. Isp will block your conection if many complaints get unanswerd |
| | Decision | What about really problematical content? |
| | | Is this hosting bulletproof? If i was a fucking idiot and wanted to use ZeuS , could i use this hosting for my c&c ? |
| | Implementation | Interested in shared hosting in such countries: China, Singapore, Malaysia, Honk Kong, Iran Packages and prices please |
| | Confirmation | I can vouch for LoyalNet, I have a BP domain from him, and even after god knows how much abuse reports, it's still alive and kicking. |

In the first innovation, most members who contributed to the discussion were skeptical, while at least one member stated that he would like to utilize the technique. However, none of the members admitted to using the technique, so it is unknown if the technique was adopted within the wider membership of Community C. The second innovation was more widely accepted as the service being advertised was traditional bulletproof hosting and some of the members indicated that they would like to utilize it, even though a few members were unsure of the true bulletproof nature of the service. One member stated that his encounter with the service was successful and that he continued to use it due to its effectiveness. One interesting observation was noted in this thread, and in others regarding bulletproof hosting observed in Community C. If the seller of the service was noted as being Russian – either through his speech, or because he also advertised his services within higher-end Russian-language communities – the members of Community C assigned him more credibility and the image of the member was enhanced. This may be because Russian computer crime is seen within the hacker and security worlds as being more sophisticated and professional (Broadhurst, Grabosky, Alazab, Bouhours, & Chon, 2014). Member C1 added that the current administrator of Community C is Russian, and Member C2 stated that most of the advanced malware comes from Russia. Despite the low number of overall postings, it appears that this technique has been adopted by the members of Community C who choose to employ it.

#### 4.2.4.4 Community C – DDoS Attacks

The third innovation studied within Community C was that of DDoS attacks, an innovation examined in both Communities A and B previously. It was observed that DDoS attacks were more accepted in general within Community C than within Community B, but still not as enthusiastically as was observed within Community A, and certain types of postings about this technique could be associated with a reduction of image. The overall volume of conversations about DDoS was also higher than the previous two innovations, as is detailed in Figure 17:

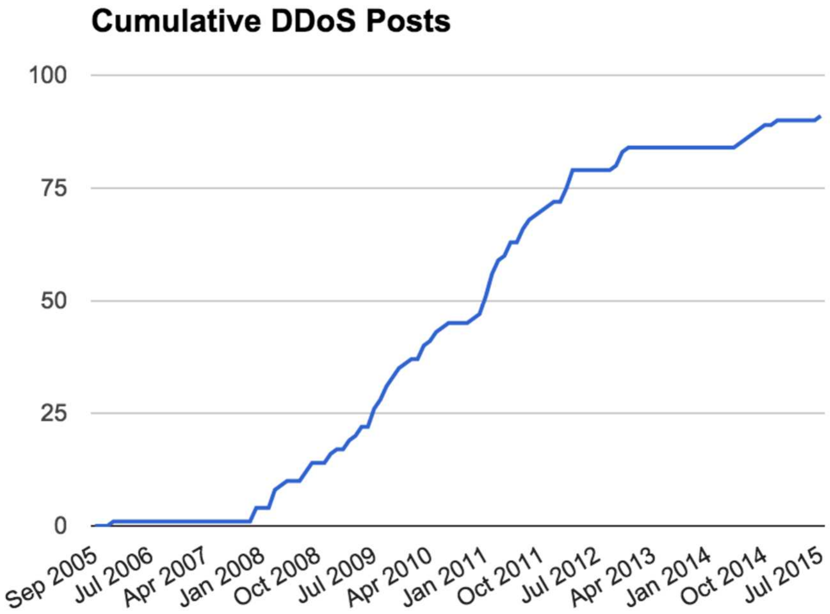


**Figure 17: DDoS Postings in Community C**

A review of innovation-specific discussions regarding DDoS within Community C revealed a significant amount of activity surrounding this innovation, much more than had been observed for the prior two innovations within this community. The curve described in Figure 17 above resembles an S shaped curve described by Rogers (2003), seen in some of the innovations described previously in this study. Despite this level of interest, there was definitely disdain observed for the technique among members of Community C, as described in the below quotes from one of the earliest DDoS threads in March 2008:

- *Ddos is lame, botnets are lame so gtfo?*
- *Only ddos attacks (gay) that are even worth attempting are distributed smurf (broadcasting using spoofed IPs etc / Syn/Ack method) or an at least 5,000 zombie net with a decent UDP flood. For sites, Just hit it with proxys. Any site*

*with sql or php is instantly fuct. But like these guys said ddos is pretty lame especially when its your site being ddossed.*

- *also with servers getting more powerful soonish ddos will be obsolete in the fact that server are/will have tonnes of bandwidth and active counter measures to monitor/detect/and reject ddos attacks.*

Member C1 observed that the public disdain may not match the private use of the technique, and stated the following:

> *In my early days DDoS attacks were sort of a grey area, when I say that, I mean my group of friends didn't approve of them but we were dabbling with botnets at the time because we were fairly new to "the sceen" and it was all the hype back then. I think I had about 60 bots at one point, just to try it out, obviously not enough to do any harm. As a user I'm really interested on how ddos attacks function, depending on the type of DDoS an individual bot can have some really interesting code. You can cause a host to timeout by never finishing a handshakes (connection routines), or even crash by sending malformed packets and so-fourth. There can be some really cool things going on behind the scenes. As a Moderator we currently have rules in place to prevent people from posting/selling bots, although we allow posting a bot's source code, and discussing methods of attack is okay. This is actually a good prevention measure as lots of new users can't code or compile source, yet want to run a botnet used for malicious purposes.*

This comment shows that the community as a whole dislikes the technique and discourages unintelligent use of DDoS, such as simply posting bot binaries or selling bots, but does encourage learning and innovation of the technique in general.

Correspondingly, despite the dislike for the technique, there was some amount of innovation surrounding various DDoS malware programs introduced by members of Community C. Table 23 describes the innovation factors observed for two such innovations:

**Table 23: DDoS Innovation Factors in Community C**

| **Date** | **Factor** | **Sample Quote** |
|---|---|---|
| December 2007 | Relative Advantage | Minidos is a botnet i unno how people feel about botnets on this forum just tell me a suggestion on it |
| | Compatibility | i think botnets are interesting.. i mean, in order to be successful in spreading, they have to use at least a couple good undiscovered exploits. also the engine has to be smart, ie not infecting the same users twice, avoiding some hostnames (AVs, .gov etc). i'd like to make a bot. :p |
| | | I get what your saying i can make a irc version to.The frist minidos was irc.But people kept crying cause there server kept going down.So that the reason why i went with Reverse Connection.And yes it is a vb app. But i always have a delphi version to.And the part about ddossing yourself.No i never had ddoss myself the older version i have over 500 bots.I have the option on there for Pause to send the commands so it wont really lagg your or nothing. |
| | | I trash the Yahoo Raw voice one i just start making it for someone that couldn't forward ports.i added more then ddossing to the new minidos.making it a small rat as well steal firefox passwords.explorer password.file transfering.desktop capture and a few other shit. |
| | Complexity | looks like visual cpp to me, but i may be wrong.. why didn't you just go with a bot that loads in IRC? too mainstream? don't get me wrong, i like your idea, but an IRC server could definitely handle more bots. |
| | Trialability | I would like to hear more idea,and i know there a way out there to use perl to have it host the bots one of my boy was using that method.But i might make this a public project.So if you like to be a part of Minidos message me.We'll go from there.oh and i forgot to say this is a Vb app.I never really got into delphi i guess you can say |
| | Observability | Sad thing about this old build is that it's really old and get detected.Another is.It's alot more advanced sence that old build.So everyone have fun with the old code.As i have seen from the source code [redacted] was the one that suppose of done this. |
| | Image | Jealous of what? Some lame Yah-Skiddies that have the ability to (in the lamest way possible/No FWB or anything) ddos websites and hit ppls IP's with lame voice bots labled as 'zombies' and think were really cool while we sit on our ass all day at mommys house. You are an idiot. |
| May 2011 | Relative Advantage | Selling the "ISR Trinity Bomb DDoS Tool" I only sell the Tool, no Hub/Bots or something else! Price will be 100€ Ukash / 125€ Paysafecard PM me, after that you get a link. |
| | Compatibility | If this is the 80kb tool, I have it as well and I can vouch. I can post some screens if you guys want. |
| | Trialability | just amazing send the bucks 4 mins ago and can play now around with isr, looks like a bomb im suprised. greetz |
| | Observability | i get the tool thanks really much the tool looks really good and strong i just tested it fast and for that price is it epic!!!!! |

The above two innovations were quite successful in Community C, with a number of positive comments and declared intentions to adopt. The first innovation, “minidos”, was eventually reinvented by being incorporated into a number of other bots that were released on Community C, for years after. There were some postings made regarding the reduction in image of using a bot to engage in DDoS, but these were in the minority. Complexity was also seen as a potential issue for minidos as the implementation did not use a less complex, but more flexible, IRC method and the member appears to be concerned that the author was adding complexity for now apparent gain. In the second innovation, a vendor was selling a DDoS tool and members were impressed with the ease of trialability and observability that allowed for rapid experimentation.

Both of these innovations, and the DDoS technique in general, were introduced in response to a particular hacking need – a desire to employ DDoS against victims, similar to what had been described in Community A. Member C1 stated that the origin of many DDoS techniques may stem from attacks on game servers by hackers, as he had “heard it was pretty common to lag rival gamers out of games”. Community C, like Community A, was a community where such attacks were desired and innovations in this area were warmly accepted.

Table 24, below, describes the adoption phases transited by two different DDoS innovations - a DDoS botnet referred to as “Silly Bot” that was developed by a member of Community C, and a DDoS service available for rent:

**Table 24: DDoS Adoption Phases in Community C**

| Date | Phase | Sample Quote |
|---|---|---|
| February 2012 | Knowledge | [initial post] |
| | Persuasion | Why still using IRC? Its an old Protocol and there are better possibilities to manage your Botnet ... Just Be creative and stop copying Bad ideas of others ;-) |
| | | Your bot has some really nice features, bin size and all, have you ever considered changing protocol to HTTP with the same binary features? that would be cool... |
| | Decision | thanks for making this Bot, i am trying it. |
| | Implementation | Bot connects and then ping time out.. Doesn't respond to any command.. Yes I have set my host to match the given host at build.. Win XP SP3 32bit (on vm) |
| | | I can't (for some reason) make the bot connect to my IRC server. mIrc connects perfectly fine but silly bot doesn't. Which options are required and which are not? |
| | | Thanks for the Update guess im first to try =). Would love to see the torrent seeder added but i can wait. Great Release !!! |
| | Confirmation | Nice job bro working fine ! |
| March 2012 | Knowledge | [initial post] |
| | Persuasion | ddos this it's [Community A] website 127.0.0.1 |
| | | free test 5 minutes, only for serious clients! |
| | | i like this |
| | Decision | I don't understand why you are trolling him, this is a legitimate service and the cracked DJ5 doesn't have SYN and UDP flood. |
| | Implementation | Can confirm it, just completed a request. He is probably genuine. |

The first DDoS innovation, Silly Bot, received a favorable reception on Community C and members were observed implementing the tool and then confirming their usage. While some individuals had difficulty with the technical details of the implementation, most members appeared to successfully use the tool. Fewer feedback was observed for the second innovation, the DDoS service – initial posts were skeptical as the poster stated he was using a botnet known as DirtJumper which had been known to be cracked and was therefore non-exclusive, and another member made a joke about Community A, possibly suggesting that the poster belonged in that community instead. Another member vouched for the poster and claimed that it had not been cracked as the public version of DirtJumper did not utilize a particular DDoS flooding method, and in the implementation phase a member did try the service and confirm that it worked. No confirmation or other feedback was observed about this service, possibly because as a service it was less well suited to the members of Community C, who were more interested in tools and code.

As described in the bulletproof hosting innovation, Russian influences were also seen as image enhancements for DDoS innovations, as observed in the following quotes:

- *I haven't tested any of these bots, but it's very unlikely that they'll be like stuff sold on [Community A], they're all from Russian origin.*
- *Oh good he is russian, so I trust him 100%, not like some [Community A] kids using booters*

Russian-speaking individuals were more highly regarded as skillful coders, and DDoS services offered by these individuals were prized over similar offerings geared towards script kiddies. The above quotes also described the theme of animosity felt towards the members of Community A, as was observed in Community B. Based on the innovation activity observed within Community C, it appears that DDoS was adopted by this community, despite the reduction in image sometimes associated with it. In all of the above DDoS innovations, support from senior members was observed to be helpful, but not critical, for favorable impression and overall adoption of DDoS innovations, such as the senior members who defended the DDoS service described in Table 24.

#### 4.2.4.5 Community C – Crypters

The final innovation studied within Community C was crypters, the purpose of which, and the need to overcome IS security countermeasures that it fulfills, had been described previously in Community B. This innovation had the most activity out of any of the innovations studied within this community, which are described in Figure 18:

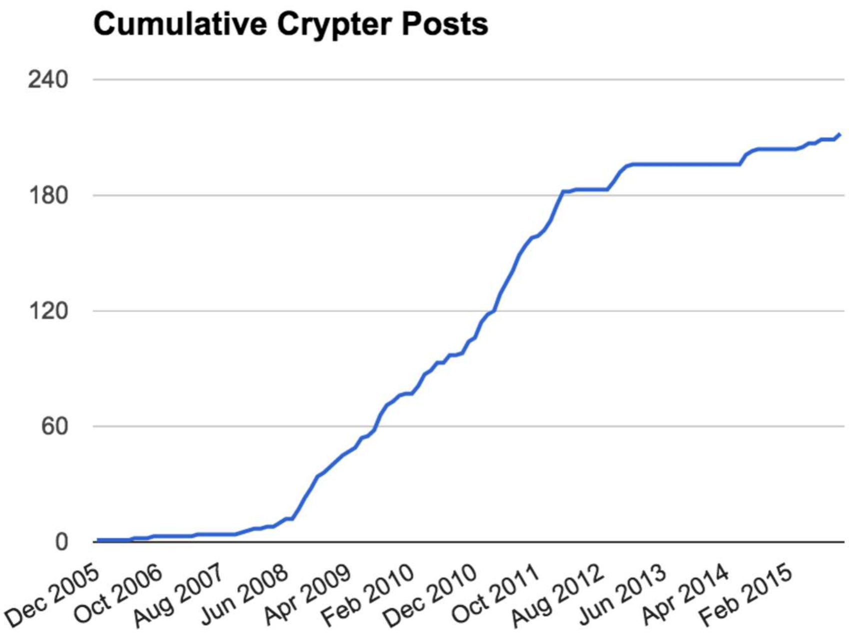


**Figure 18: Crypter Postings in Community C**

Figure 18 strongly resembles the S shaped curve predicted by Rogers (2003). The amount of interest in crypters is logical given the community's focus upon malware – encrypting malware is a key component of distributing it and ultimately infecting victims. According to Member C1, "When Antivirus company's started implementing heuristic based detections lots of previously undetected Malware was tagged. Malware that gets quarantined by AVs is pretty useless, so you have two options, you can re-code the entire piece of malware, which then may still be detected by heuristics or, you can obfuscate it." As such, a strong hacking need predated the introduction of the technique into Community C.

Member C1 argued that there are also legitimate uses for crypters, such as protecting programs from piracy, but added that many crypters have a bad stigma attached to them due to their prevalent use within malware. Again, there was an observed emphasis upon learning from Member C1, who added that “there are really cool innovations that can go into them though such as executable forking which I was around to see the evolution of. We try to promote posting source code so that others can learn how the stuff works”.

The activity regarding crypters within Community C was a mix of public code and vendors selling compiled crypting programs or online services. Table 25 describes two such crypting innovations and the innovation factors that played a role in the decision to adopt:

**Table 25: Crypting Innovation Factors in Community C**

| Date | Factor | Sample Quote |
|---|---|---|
| September 2009 | Relative Advantage | since Octopus 1.0 received good feedbacks from customers (see thread), I'm here to show the new version, 1.1… The client will crypt the binded files with the chosen encryption key (generated and randomized automatically on client execution anyways, but also customizable). File/s code will be completely scrambled, and Fully UnDetectable. |
| | Compatibility | This "EOF Support" that most common crypters use, what is it? Does it put the binded binary in the end of the file or? |
| | Complexity | Not that it's important or anything, but could you not just simply choose to not encrypt the PE-Loader and instead of storing settings in the file at some offset, change some strings from the resource file? If you do it correct, the PE-Loader does not even have to be encrypted. As an example take Cryptic. I used something I posted here a while ago and made it 100 % UD. |
| | Trialability | I saw source, i vouch for [redacted]. Its not ripped, all is in credits which he used in snippets. |
| | Observability | I bought 2 weeks ago.Very nice software and very good service, especially i like usb spread very much. |
| | | I can say "encryption key" (Semi-polymorphic code) really woks.Yesterday i have noticed my server was detected by quick-heal,then i crypted the server again with different encryption key then its 0/23 again. |
| | Image | AHHA admins.. why are you defending this guy? do you know him? NO so pleas don't ban [Redacted] is ripper and i will repeat everydays ! FUCK HIM i know when he started and it was since 5/6 months and he were knowing NOTHING no knowledge. |
| September 2008 | Relative Advantage | Here the second release of the BiohazarD Crypter: This time ihave added anti-Sandboxie,Anti-VirtualMachine,EOF Support(works with Bifrost!) Prockiller,AntiDebugger,Pack Server and XOR Encryption are not added completely.... I started to Sell my Crypter: |
| | Compatibility | [Community C] isn't a online payment for crypters !!! Your crypter is coded in vb6 so no compatible vista and there are a lot of bugs It dont work !!! |
| | Trialability | I am go test it now! And yes he is a P R O - C O D E R \| Delphi, Dev C++, VC+, VB and PHP |
| | | ok the test with Poisonivy Server: Before: NoVirusThanks.org - Online Virus Scanner Engine version 1.2-build[090908] After: NoVirusThanks.org - Online Virus Scanner Engine version 1.2-build[090908] |
| | | connect to msn [redacted] pls i will see if i can buy the new one and i want you to contniue tuts plz :d |
| | Observability | hum yes... GUI work good, nice work but your application GUI isnt autonomous Ocx Size : 1.6 mo looool => very big And for the final package, it seems to work well 7.5/10 => good crypter |
| | | this shit is FUD since a few days XD |

Both innovations above were eventually adopted by Community C despite some initial concerns regarding compatibility and complexity. Additionally, few mentions of image were seen in most innovations, with the exception of an individual who was attempting to "flame" the thread for malicious purposes. Some members expressed concerns that some of the crypters may have been "backdoored" with malicious software planted by the developer, but the observability of the results of the innovation helped assuage those concerns. Overall across all of the crypter-related innovations seen in Community C, the most critical factors were seen as compatibility and trialability, with some concern for image in two key areas – calling members who tried to sell innovations rippers (related to trialability, so potential adopters can verify that the product works before purchasing) and questions that telegraphed that the member was a potential "n00b", or newbie. The reaction to questions from "n00bs" about crypters were public shaming and a reduction in image for that member. However, one posting in particular was observed where a member who identified herself as female was asking such questions, and the (presumably male) members were enthusiastic in offering assistance, which may be an interesting side observation around the lack of women in hacker communities.

Many of the crypting innovations observed in Community C also transited the innovation-decision process as members evaluated the innovation factors, and example innovations are described in Table 26:

**Table 26: Crypting Adoption Phases in Community C**

| Date | Phase | Sample Quote |
|---|---|---|
| September 2008 | Knowledge | [initial post] |
| | Persuasion | Thanks for this great share, going to look at the source. |
| | | thank u it's a very good crypter |
| | Decision | You are the man! Will credit you FFS =D Didn't even believe this was so easy achievable. However, to make it more UD can I just mess around in the Rc4 encryption? |
| | Implementation | Very nice tool mate. It is very good and very simple. |
| | | I've got it UD, only NOD32 and Avira detect it as heuristic, is it because of the API's used? Someone can give me examples how to call the API's with CallApiByName ? Thanks |
| | | yes tested with antivir UD stub but when i crypt it's detected :\ |
| | Confirmation | And that is certaintly annoying! Still he made it FUD? And price? Fuck it feels dumb to not being able to take a look yourself =/ Well, if I get my own crypter programmed, I'm gonna credit you for this example! Thanks again =D |
| | | It was detected by 12 then I messed about with the encryption and got it down to four, although it says that 'Is is the trojan dropper'. Now I obviously though that removing the antitrojandropper would work but that didn't what do you guys think? |
| | | now HOW DO YOU MAKE THIS MOTHERFUCKER FUD????? goddamit, and how about in ENGLISH??? goddamit! tell us to change code around, there are 500000000 lines of code, you want we should change all that around??? im about to fucking quit, its 5 am and i have accomplished NOTHING! this is BULLSHIT!!! fuck it, i fucking quit. |
| September 2009 | Knowledge | [initial post] |
| | Persuasion | nice gui good functions to bad u sell it |
| | | Very good [redacted]. I wish you luck and i hope this tool stay FUD for a long time. Good sales for you. |
| | | mmm, how about fiding you're proper name for this ripped project? i know that maybe you recoded it but the whole code is public so it's pointless to sell it.. do w/e i don't think you will find any customer ... guys if you want to buy any crypter/binder buy iCrypter it's the one that you're sure it's working and coded by the person who sell it. |
| | Decision | I would be interested i9n this software. does anybody has tested it and can give feedback? |
| | | I can give you feedback. [redacted] showed me it through TeamViewer. It's really nice. And still FUD. Lots of options, all working. |
| | | why should they buy a shitty ripped crypter? buy Icrypt or brmcrypt the longlast fud but i suggest BRM Crypter why?: stub in C++ and will last long FUD + more functions and for 45$ |
| | Implementation | Is good , i have it already ..................... |
| | Confirmation | I have tryed this tool and must say it is not bad at all, at the time I get it, it was 100% FUD (NVT) Crypting PI 2.3.2 Spreader was working (only thing is it was not working only on USB but infect ALL drives). Well the tool does its job pretty well... |

In the first example, the author – a senior member of Community C – posted a crypter for free for others to experiment and learn from. Other members appeared appreciative and supportive in the persuasion phase, and numerous members transited the decision phase and adopted the innovation. Many of these members had technical questions regarding implementation, and later confirmed that they planned to reinvent the innovation by incorporating it into their own crypters and "crediting" the original author. This crediting appears to be a way to show respect for the initial poster and elevate his status within the community. Also in the confirmation phase, two members expressed difficultly obtaining "FUD" – one of the members was able to pass eight virus scanners, but the second member, who was relatively unskilled, was unsure of how to proceed. The other community members attempted to help him, but it seemed unlikely that he would continue to use the innovation, although the other members who weighed in on the thread appeared to indicate that they would continue use.

The second example was of a vendor, a senior member in Community C, who was selling a crypting service to other members. Most of the reception to the service was positive, with the exception of two other senior members who made a number of disparaging remarks while they promoted other vendors who were selling competing services. The objection of these members was that the technique behind the service had been "ripped" from another member. However, other members on the forum defended the original author and supported the service, and a number of members indicated that they planned to adopt, or had adopted and were satisfied with the service. The comparison of these two crypting innovations – one offered for free, and the other only available to paying customers – demonstrates some of the barriers to adoption that are faced when financial motivations enter into the adoption process. The fact that the innovation was being disparaged and other services were being promoted may indicate that the members were spreading unfavorable opinions because they would benefit financially.

#### 4.2.4.6 Community C – Summary

A review of activity within Community C revealed a very different community of hackers than had been observed in Communities A or B. Community C was focused exclusively on malware, and specifically around the authoring of malware. As a result, some innovations which were more popular in other communities – such as SQL injection, or DDoS – were less active here, while others, such as crypters, were much more active. Discussions around innovations were much more technical and were focused upon code-level improvements. Like the other communities, image was seen as quite important, but the increase of image from innovations associated with Russian members or vendors in particular was observed. An innovation that was associated with a Russian origin was seen as much more successful and appeared more likely to be adopted, because of the positive image associated with Russian cybercriminals. While image was seen as important, there was significantly less derision observed for "script kiddies" or for less-skilled techniques, such as Havij, that had been observed within Communities A or B.

Complexity was also an important factor, with more complexity associated in general with increased support for the innovation. Innovations that were seen as too simplistic were often not adopted. Member C2 emphasized this point by stating that high-level languages such as Visual Basic or .NET were unfavorable and when they were observed in malware, those threads would often be moved to Community C's "hall of shame". Member C2 also stated that Russian (as well as Chinese) malware programmers favored lower-level languages and this was one reason for the success and popularity of Russian innovations within the community. The skill level of Community C was observed to be high, due to the amount of innovation activity and the technical aptitude of its members, who were actively utilizing these techniques to engage in attacks.

Unlike Communities A and B, many of the innovations proposed within Community C were actually developed by members. These innovations were introduced to the community in response to pre-existing hacking needs of members that were in line with the malware-oriented focus of the community. The focus upon software development of malware led to actual code-level discussion of innovations and led to more members proposing technical innovations. Similar to Community A, the focus of the forum appeared to be upon all three modes – mostly upon *operations* of attacks, with some vendors *selling* services or tools and with some *learning* regarding how attacks work. Also unlike the previous communities, senior members played a much smaller role as change agents – even though there were differing levels of seniority within Community C, their impact was minimal, and innovations appeared to be evaluated primarily upon their merits, or upon the impacts of image as described previously. Member C1 stated that postings by senior members get more positive feedback than postings by newer members, but also that these postings are of a higher quality because the senior members have "a reputation to keep up". Senior members, in the opinion of Member C1, have also been around longer and are more experienced at coding, and therefore are in a better position to introduce new innovations.  Both Members C1 and C2 stated that status matters little when a good idea is proposed – if an excellent innovation is proposed by a new member, it will still be supported and adopted.

As in the prior communities, the communications channels employed by members appeared to be primarily public postings and private messages, with some reliance upon outside chat services and websites.  Member C1 stated that private messages in particular were often used for the negotiation of sales between sellers of hacking tools and their buyers.

### *4.2.5 Case Study Analysis – Community D*

#### 4.2.5.1 Community D – Overview

The fourth and final community studied as part of this research was Community D. This community was founded in 2013 and was a very large forum, with over 270,000 user accounts created.  Of these user accounts, about 43,000 were found to have made posts.  While the data collection for this community was underway, it was observed that upward of 4,000 members were signed on at any given time, and the site boasted that the highest number of users that were ever online at once was 35,000.  This indicates that Community D is a very active forum.  Once a user has registered for an account, they appear have access to the entire community.

The main focus of this community was found to be credit card fraud, or “carding”. Peretti (2009) describes the underground nature of the carding forums, and highlights many of the features that were observed in Community D, including tutorials, vendors of hacking tools and services, proxy lists, and the naming and shaming of rippers and scammers. Like the other communities investigated, Community D had a hierarchy of members, with junior members at the bottom and senior members (called "Top Carders"), moderators, and administrators at the top. Like Community A, senior membership can either be purchased or awarded by the operators of the site. Community D also employed a reputation point system to help members determine the level of trust that should be placed on other members, which was quite important when negotiating purchases between members. It was noted that many, many vendors were active on this community, and a significant percentage of all postings were observed were advertisements for tools or services related to carding. Because this forum resembled a marketplace more than any other forum reviewed, reputation points were key so members could avoid rippers and scammers. While the primary language of the forum was English, a large number of participants in the community were from the Middle East.

While the focus of Community D was centered around credit card fraud, there were hacking innovations that were observed, even though the hacking activity was centered almost entirely around obtaining credit cards or other financial information such as credentials to online banking. Based upon the input of the expert panel and the innovations encountered in the previous three communities, the following innovations were selected for study within Community D:

- Havij
- DDoS attacks
- Malware-related items (including keyloggers, Zeus botnets, and crypters)
- Credit card skimmers (closely related to point-of-sale attacks)

Data collection and analysis for Community D was similar to the methodology employed in the previous three communities. As before, interviews were sought with two community members. However, despite repeated efforts, this research was unable to enlist blackhat members of Community D to participate in this study; the apparent reticence of members to speak with a researcher about the illegal activities was too great, a problem that was not encountered in the previous three communities. This may be because the focus of Community D was explicitly criminal, versus the other communities' more varied focuses upon learning, sales, or other less-illegal, but still blackhat, activities. The researcher contacted approximately 20 members; three replied and declined to participate, while the remainder simply did not respond to requests for interview. As an alternative, two members of Community D who were actually undercover whitehat IS security industry researchers were interviewed about their experiences within the community. Members D1 and D2 are both long-time members of Community D and are security researchers employed by small cybersecurity firms.

#### 4.2.5.2 Community D – Havij

The first innovation studied within Community D was Havij, as was first studied within Community A specifically, and Communities B and C more generally under the umbrella of SQL injection attacks. There was a significant amount of postings observed regarding this tool – primarily in the context of members utilizing it to exploit online e-commerce sites that are vulnerable to SQL injection attacks, in order to dump credit card details stored within the SQL database of the site. Member D1 stated that this technique is popular within Community D, and in addition to Havij the tool "SQLmap" is also used. Member D2 added that Havij was preferred over SQLmap or other tools.

An interesting trend was noted where members would post links to websites that were "hackable" with Havij and encourage other members to attack and obtain credit cards. This was a surprising discovery, as one would assume that more hackers exploiting the site would lead to the same credit cards in the hands of multiple attackers, which in turn would lead to them being identified as compromised by the credit card issuing banks and would lead to attempted purchases being marked as fraudulent.  This may indicate that these postings are done to obtain reputation points or other image enhancements within Community D and not for the cards alone.  Figure 19 describes the appearance of Havij within this community:

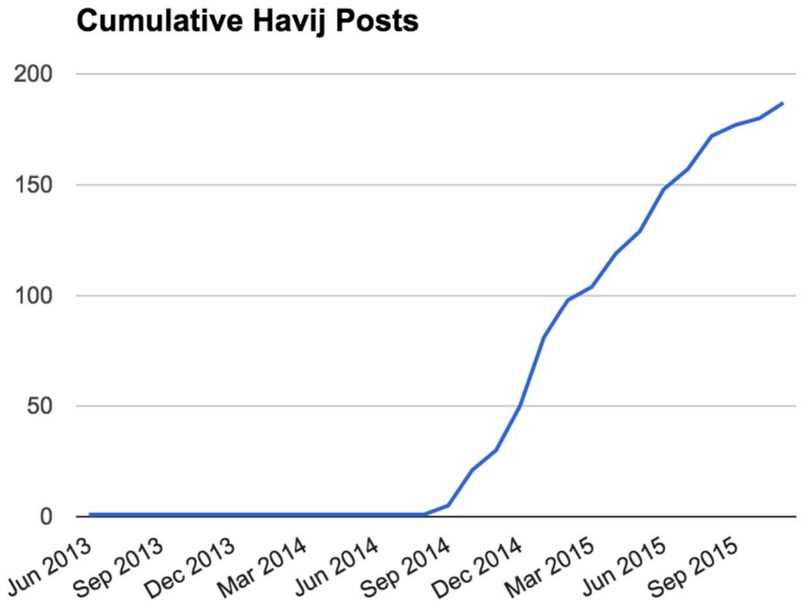


**Figure 19: Havij Postings within Community D**

A total of 187 topics about Havij were noted, of which twelve were tutorials. As observed in previous communities regarding SQL injection attacks, some participants chided members who were asking about Havij and suggested that they “do it manually” and not use shortcut tools. However, there were much less of these types of comments as had been seen in the other three communities, suggesting that the members were more interested in the results of the hacks than the way they were obtained. Member D2 stated that the members who believed that Havij was a shortcut tool were in the minority of members in Community D, with most favoring the tool due to their low skill level. Additionally, despite the fact that there was a significant amount of postings regarding Havij and SQL injection in general within Community D, there was significantly less innovation activity observed than in the other three communities. In one thread, the merits of Havij were evaluated against another tool, called “SQLi Dumper”, as detailed in Table 27:

**Table 27: Havij Innovation Factors in Community D**

| Date | Factor | Sample Quote |
|---|---|---|
| October 2014 | Relative Advantage | SQLi Dumper Features: -Suports 20 methods of SQL Injection; -Suports Multi. Online search engine (to find the trajects); -Automated search for data in a bulk URL list; |
| | Compatibility | can l get cc details with this? |
| | | I use havij in your opinion SQLi Dumper 7.1 is better than havij? |
| | | is this tool gud..??? I mean i used havji.. So is this better than havji.?? |
| | Complexity | this tool it's like havij sql injection bro i recommend this tool for advanced user, so n00b stay away |
| | | if me you are going to ask itsmore easy to use sql dumper coz its all in one.just a little bit not newbie friendly. |
| | Trialability | I have this tool, and i have uploaded it pm me I'll give you link. My download link doesnt have any passwords so youcan safely scan thedownload link. Pm me and give me rep I'll give download link |
| | | How do i instal this on windows 8 |
| | Observability | havij is a baby compare to this, this tool is fuckin amazing. i love u no homo |
| | | fabulous bro i used it yesterday results amazing.... |
| | | mine doesnt come up with any results any body know why? |

In the thread described in Table 27, there was no mention of the image enhancement or reduction from using either Havij or the SQLi Dumper tool, a trend observed in most conversations regarding this tool and a sharp difference from what had been observed in other communities.  The tool was found to be compatible with the needs of the community – these needs, namely the necessity to obtain credit cards via SQL injection, predated the introduction of the tool, and aligned with the overall purpose of the community – and the complexity of the tool was found to be high as it was described to be "not newbie friendly" and that "n00b[s] stay away".  This increased complexity appeared to be a positive factor, however, as Havij was seen as being too simplistic with less features than SQLi Dumper.  The easy observability of the results of this tool encouraged members to consider adoption, and in general it appeared that members in this thread were enthusiastic about SQLi Dumper as an alternative to Havij.  A review of additional threads about Havij within Community D, however, indicate that very few other mentions of SQLi Dumper were observed.  Table 28 describes the innovation-decision process of one discussion regarding Havij and corresponding "dorks", or search engine keywords to help hackers find sites vulnerable to possible SQL injection:

**Table 28: Havij Adoption Phases in Community D**

| Date | Phase | Sample Quote |
|---|---|---|
| February 2015 | Knowledge | [initial post] |
| | Persuasion | omg thanks i always curious about this and i didnt know the ccs can found like this respect |
| | Decision | clean file?? will try soon |
| | Implementation | dorks are great but havij doesn't work and I don't see a scanner |
| | | I only get gr3enox to work once or twice, then it stops getting results, probably due to google blocking it, any bypass? |
| | | doesnt work error message when i launch havij and greenox fuck ,.. backdoored shit |
| | Confirmation | woow nice share mate ... really HQ dorks thanks a lot ... always appreciate to see people sharing nice stuff  like all the others lol if u have some more let me know ... but thanks again !!! |

In the above thread, a member posted a file with Havij and a tool called "gr3onox" to help automate the searching for vulnerable sites using dorks. There was little opposition to this innovation in the persuasion phase, although some members were concerned that the posted file may contain a virus. Many of the members expressed frustration getting the tool to properly run in the implementation phase, and in the confirmation phase a member complimented the original poster on the high quality of the dorks but was silent on the other tools, suggesting that the uploaded tools may not have been adopted by these particular members. In general, reviews of other threads related to Havij indicate that the tool was adopted by members of Community D, perhaps grudgingly, and enjoys active use by community members.

#### 4.2.5.3 Community D – DDoS Attacks

The second innovation studied within Community D was the DDoS attack. This community contained a smaller number of postings, a total of 72, about this technique than had been observed in the prior innovation. This is likely because DDoS attacks tend not to fit the stated mission of the community, which is credit card fraud. Figure 20 describes the observed postings regarding this technique.

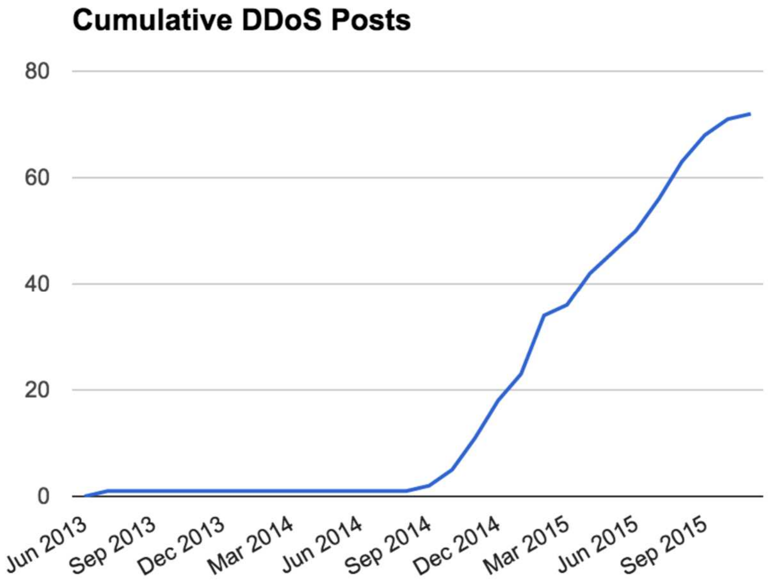


**Figure 20: DDoS Postings within Community D**

The shape of this curve is similar to the one observed in Figure 19 for Havij – an initial posting early in the life of the community, and then nothing until late 2014, when a sharp rise occurs. Most of the postings observed about the DDoS technique consisted of requests to attack a particular site or requests for where to obtain a DDoS service. In one instance, a vendor was observed advertising DDoS services (referred to as a "stresser"), but another member posted a comment regarding the skill level of the members of Community D: "Hi [redacted] they dont know what is this stresser they think something else try to sell to other forums because this forum its only for free tools for noobs :P Try to sell to [Community A]". Other requests for DDoS services were observed where members directed individuals to Community A, as that was perceived as being a more suitable place to find vendors. As a result, the hacking need for such a technique within Community D is unclear, as it does not appear to align with the stated goals of the community. It may be that there is overlap between the carding needs of the members and their hacking needs, where DDoS would satisfy the latter but not the former. Other hacking-related activities were observed that did not exactly align with the carding nature of the site, such as web defacements, so it seems that a need for general criminal techniques such as DDoS did exist, but at a much lower level than in the other communities examined.

There were very few examples of innovation activity related to DDoS within Community D. One example involving a DDoS tool available for purchase where some of the innovation factors were observed is described in Table 29:

**Table 29: DDoS Innovation Factors in Community D**

| Date | Factor | Sample Quote |
|---|---|---|
| January 2015 | Relative Advantage | I am selling a Dos tool, it's very powerful. I sell this to a limited members in forum. This is a Priv8 tool, and very easy to use! |
| | Compatibility | How much GB/PS ? is it spoofed ? Any other costs fees for it ?? |
| | Trialability | Vouch Copy please |
| | | oh best tool, can u attack demo to site for me send u? |
| | Observability | The powerful of attack depends from your internet speed, if internet speed is very strong(powerful) the doser will be amazing. |
| | | Vouch for this guy , best DoSS tool |
| | | Vouch for you bro. you really have amazing ddos. |

In the above innovation, the original poster states that he is selling his DDoS tool to members of Community D only. One member asks about the capabilities of the tool, to determine if it is compatible with his needs, while other members ask for a copy for testing or if the operator can attack a site to demonstrate the effectiveness of the tool. Other members post about their use of the tool and describe its effectiveness. Interestingly, no mention of complexity or image were observed. Review of other DDoS threads did not result in any innovations which appeared to transit the innovation-decision process, and members were not observed weighing the decision to adopt or reject DDoS attacks. Member D2 stated that more recently, members were observed seeking out "DDoS-for-hire" services. Due to the number of postings observed regarding DDoS and the fact that members were actively seeking solutions, it appears that the technique was ultimately adopted by members of Community D.

4.2.5.4 Community D – Malware

The next innovation studied within Community D was the use of malware. There was insufficient activity regarding specific types of malware or malware-related tools and services – such as keyloggers, Trojans, and crypters – so coding was conducted of terms related to malware in general to determine the level of adoption. The keywords utilized were obtained from the reviews of related topics in previous communities. Figure 21 describes the malware-related postings observed within Community D:

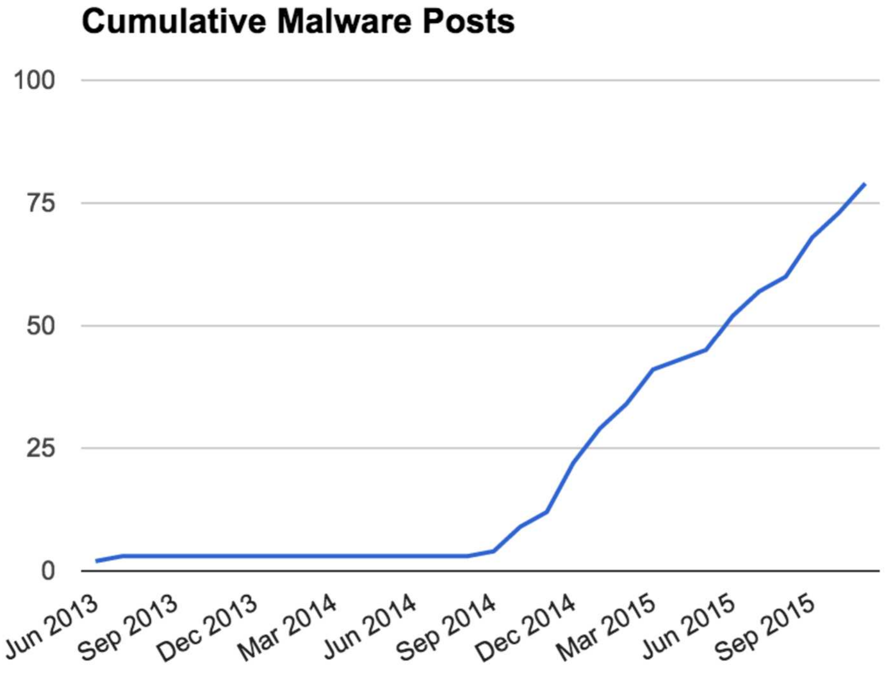


**Figure 21: Malware Postings within Community D**

A total of 79 postings related to malware were observed in Community D, of which five were tutorials. The chart above also resembles the previous two innovations observed in this community. It was initially unclear why there is early activity in 2013 but none until late 2014; according to a review conducted of the Internet Archive and other tools, the website for Community D was active during this time and no error messages appear to have been displayed, so it is possible that the site may have suffered a loss of data or just fell out of vogue among hackers during this period. Member D2 provided background, stating that many members were active on a sister community of Community D, which was shut down in late 2014 and those members migrated to Community D.

While there was a somewhat high level of interest in malware and associated techniques, there was very little innovation activity observed. Malware as a technique certainly fit a pre-existing hacking need and aligns well with the overall mission of the community – to obtain credit cards or other financial information. Member D1 stated that malware activity within Community D tends to involve keyloggers or Remote Access Trojans (RATs), but added that he has not seen any "high reputed" malware in the community, suggesting that the skill level is low. Member D2 added that "I wouldn't say there was exactly a thought leader when it came to malware. No one really posted more than anyone else did on the subject". Only one thread related to malware was observed where confirmed innovation activity took place – and this thread was a member selling a tutorial for spamming, as described in Table 30:

**Table 30: Malware Innovation Factors in Community D**

| Date | Factor | Sample Quote |
|---|---|---|
| March 2015 | Relative Advantage | Selling: Full Spamming Tutorial [A-Z] Pics Included Q. Why should I do Spamming? A. If you hate to buy CC's, Bank Logs, Accounts etc. you'll need to start Spamming so you could get your stuff by yourself. Also if you need to spread your Keylogger or Malware, Spamming will help a lot. |
| | Compatibility | Good guide, is original and not some out dated copy pasta. Weary helpfull if ur new to spamming. To be honest I don't want tis guide to get out too much cuz it's so helpfull |
| | | I've added some improvements to the guide It's still on sale and really good guide if you looking to learn Spamming. |
| | Complexity | Good tut for beginner and well explain nice guy with good sense of humor and good communication....i bought it and i like it....$$$ well spent A+++++++ |
| | | it is good guide only for newbie in my opinion. all steps are accurate and it is clearly explained. does not include any advanced techniques though. |
| | Trialability | Vouch Copy please |
| | | Vouch Copy only for Staff member |

The table above describes the member who was selling the spamming technique tutorial, with an emphasis upon using the spam to spread malware or attack organizations. For the compatibility factor, one member describes how the tutorial fit his needs by being original, and the original seller states that he's added even more content to help members with their spamming needs. The complexity of the innovation was low, which was a helpful factor for selling to newbies. Some members asked for free "vouch copies" so they could test the innovation and describe their results, but the seller stated that these copies would only be provided to staff members of Community D. The seller did end up providing copies of the tutorial to senior members, who vouched for the content, and subsequently other more junior members began to purchase the tutorial and added their own vouches. No mention of the observability of the seller's techniques were observed, and as had been noted in the previous two innovations for Community D, no mention of the impact upon the image of members was noted.

Within this same thread, parts of the innovation-decision process were transited by some members, as can be observed in Table 31:

**Table 31: Malware Adoption Phases in Community D**

| **Date** | **Phase** | **Sample Quote** |
|---|---|---|
| March 2015 | Knowledge | [initial post] |
| | Persuasion | Ok, good method, well explained with images and important tips for newbies. good luck with sales. |
| | Decision | when are going to sell at a discount again, am interested |
| | | I'm quite interested in what you've got to say - if you could pm me your jabber, I'd like to purchase and have a convo with you. |
| | Implementation | vouch for this guy ! i get the method ! |
| | | I bought the last copy, it is simple and easy to understand, my vouch for him. |

In the knowledge phase, the seller made Community D aware of his innovation through his initial posting to the site. In the persuasion phase, a senior member who received a vouch copy of the tutorial weighed in and added his support to the seller. In the decision phase, two members appeared committed to adoption and asked about purchasing the tutorial. And in the implementation phase, two members stated that they had purchased and that the tutorial was a success. No observation was made of the confirmation phase, where members could decide to continue or discontinue use; however, that makes sense given that this innovation was a tutorial, as the initial use of the tutorial should be sufficient to gain the knowledge conveyed in it, and further uses would not be necessary.

Based upon the total number of postings observed related to malware within Community D, it appears that the community has adopted the general concept of malware as a means to obtain credit card information.

4.2.5.5 Community D – Skimmers

The final innovation studied within Community D was the concept of skimmers. Skimmers are hardware devices designed to steal credit and debit card information by reading the magnetic data stored on the card, and can be placed by criminals into ATM machines or swiped surreptitiously during credit card purchases (Sharma & Rathore, 2012). Skimmers are closely related to the concept of point-of-sale malware as discussed by the expert panel, which surprisingly had no measurable level of activity within this community. One would, however, expect there to be a significant amount of innovation activity regarding the development of skimmers in a forum such as Community D. This was not the case, and Figure 22 describes the skimmer-related postings observed in this community:

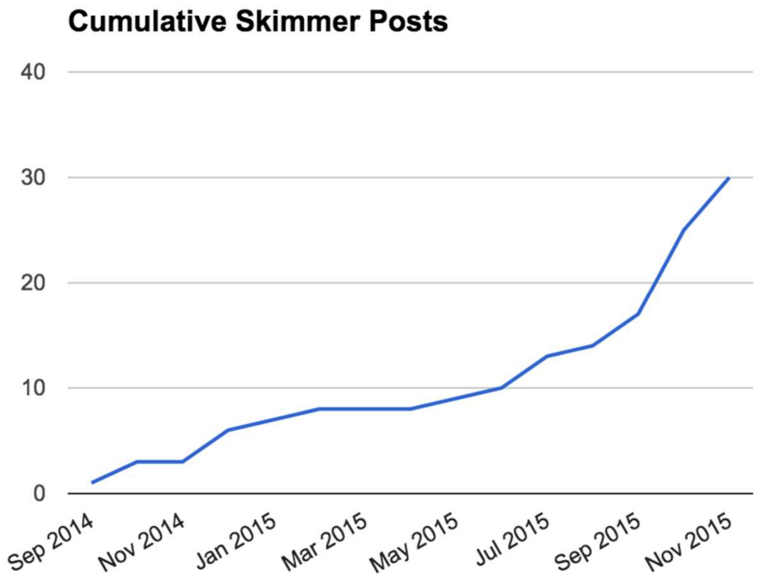


**Figure 22: Skimmer Postings within Community D**

A total of 30 postings regarding skimmers were observed within Community D. Almost all of these postings were either vendors selling skimmers, or members wishing to buy them. The need for this innovation is clear – skimmers are an easy way to obtain credit card data and this feature aligns perfectly with the overall nature of the community. There was only a single observed posting where skimmer innovation occurred – a member posted that his skimmer was failing on cards obtained in Europe and was asking for ways to improve it, and where his cards could be cashed out. This was a collaborative thread, however, and no experimentation with the member's innovation was observed, so it is not a good candidate for documented adoption activity. It does appear that the members of Community D have adopted skimmers as a technique, given that a number of individuals sought to purchase and sell them, but almost no innovation activity was observed for this technique. Member D1 stated that he has observed very little innovation regarding skimmers in this community, and has observed hardly any discussion of emerging credit card countermeasures such as "201 EMV", or computer chips embedded in cards. Member D2 stated that the limited amount of postings observed for skimmers related to ATM or gas station skimmers, but little else.

#### 4.2.5.6 Community D – Conclusion

Community D was observed to primarily be a marketplace of relatively low-skilled hackers who were focused exclusively upon credit card fraud. Member D1 stated that few members are "actually wildly skilled" – most of the members are involved in simple pickups of cash orchestrated by more skilled hackers, while only a few members actually possess the skills to develop tools and infect victims with malware. Some innovation did take place within this community, but it was quite limited with most of the focus of the community upon buying and selling tools, techniques, or services related to carding.

With the exception of DDoS, the studied innovations did meet a pre-existing hacking need, and DDoS appears to be tied to a weaker need for criminal techniques generally. The roles of senior members appeared to be similar to those observed in the previous forums – Member D1 related that vouches from senior members boost sales, as the reputation of these members "automatically generates interest" of members who observe the vouch. Member D2 confirmed this, stating that postings by administrators, moderators, or other senior members "could really drive a lot of the conversation in the forum and were certainly revered by others". Similarly, Member D2 added that "he biggest difference between a n00b and a l33t member of the forum proposing something are more in the number of replies in thread and consideration that is given to an idea", and that "older members are more likely to be thanked/repped for their contribution", confirming the similar level of importance of these senior members.

For the innovation activity that was observed, it was found that the role of image played no part whatsoever, unlike the prior three communities studied. This may be because the focus of each forum differed significantly, and the effect of an innovation upon the image of a member within the community may be unimportant in an almost purely marketplace environment. What was important to members was reputation points, to avoid scammers and rippers; however, the innovations themselves had no impact upon the reputation points of an individual member. The other innovation factor which was under-observed was that of trialability; the reason for this is unknown, but it is possible that the membership of Community D was uninterested in experimentation and trying innovations before deciding to adopt. As a result of these observations, the overall skill level of Community D appeared to be low.

Of the four innovation types examined, only one – a DDoS service – appeared to have actually been developed by a member of the forum, and DDoS arguably had the least to do with carding of all the innovations. The communications channels employed by the members of the community matched others observed in the prior communities, and consisted of public postings, private messages, and outside chat services such as ICQ or Jabber favored by individual members. External websites appeared to be less important, as the vendors selling wares tended to sell directly within the forum and did not direct buyers to outside websites where tools or services could be purchased. Unlike all three previous communities, the community did not have an emphasis upon learning, but instead purely upon *sales* and upon the *operations* of attacks.

### *4.2.6 Cross-Case Analysis*

As described in Figure 3, the next phase of this research involves cross-case analysis. To some extent, this has been happening already, as each of the above cases was influenced by the findings of the previous cases, as had been described in the explanation-building analytic procedure outlined in the analysis of Community A; however, further conclusions can be drawn across cases. Additionally, reviewing the results of these cases in light of DOI theory will also reveal additional insights.

To begin with, during the analysis of each case it was observed that each community had different focuses, with different activities occurring in each. Table 32 summarizes what was observed for each of the four communities examined, including the overall focus of the community and the types of activities that were observed, in order of observed priority for the community. The observed skill level was also provided for each community.

**Table 32: Summary of Communities**

| Community | Overall Focus | Skill Level | Activities (by priority) |
|---|---|---|---|
| Community A | General hacking | Medium | Operations, sales, learning |
| Community B | Education | Medium-High | Learning |
| Community C | Malware | High | Operations, learning, sales |
| Community D | Carding | Low | Sales, operations |

Interestingly, the skill level for Community A was determined to be medium, despite the active disdain for that community from Communities B and C. This is because the skill level for Community D was found to be exceptionally low, and by comparison Community A was determined to be a level above. Each of these communities evaluated a different subset of innovations that were recommended by the expert panel. To assist in the analysis of these cases, it would be helpful to first summarize the tool and technique innovations that were adopted or rejected by each of the communities, as described in Table 33:

**Table 33: Summary of Community Innovation Adoption Decisions**

| Community | Innovation | Decision |
|---|---|---|
| Community A | Havij | Adopted |
| | Amplification Attacks | Adopted |
| | Bulletproof Hosting | Adopted |
| | Shodan | Adopted |
| Community B | SQL Injection | Adopted |
| | DDoS Attacks | *Rejected* |
| | Crypters | *Rejected* |
| | Bitcoin | Adopted |
| Community C | SQL Injection | *Rejected* |
| | Bulletproof Hosting | Adopted |
| | DDoS Attacks | Adopted |
| | Crypters | Adopted |
| Community D | Havij | Adopted |
| | DDoS Attacks | Adopted |
| | Malware | Adopted |
| | Skimmers | Adopted |

As can be seen above, the four hacking communities reviewed adopted most of the studied innovations, with only three innovations clearly rejected.  When considering the adoption or rejection of these innovations, note that the technique innovations should be considered as innovation categories – one technique innovation, such as DDoS attacks, may have a number of tool sub-innovations, such as Community C's "DirtJumper", "minidos" and "Silly Bot".  These tool sub-innovations can be adopted or rejected separately without affecting the adoption of the overall technique, in this case DDoS. Examining the contrary cases of why an innovation was rejected will also prove useful, as rejected innovations represent instances where the innovation-decision process breaks down.  Note that we are only counting an innovation as rejected if the entire community appears to reject the innovation, not individual members.  Even when an innovation is widely disparaged, such as Havij within Community A for image purposes, if that innovation experiences a significant amount of innovation activity and appears to be adopted by the majority of participants, for the purposes of this research the innovation will be considered adopted.

The first rejected innovation in Table 33 was that of DDoS attacks in Community B. This was a passive rejection per Rogers (2003), and the only passive rejection observed during the study.  DDoS attacks were examined in each of the four communities - in Community A under the specific subsection of amplification attacks, and in the other communities as DDoS more generally.  This technique was adopted by Communities A, C, and D, but rejected by Community B.  Within Community B, the primary reason for this rejection was due to the negative image associated with DDoS attacks; this class of attacks was strongly associated with "script kiddies" and members who attempted to initiate discussions about DDoS were publicly shamed.  Strong social pressure was found within Community B to not engage in these attacks, and members risked having conversations moved to the "Board of Shame" for public ridicule if they promoted the technique, although some small amount of discussion about DDoS defenses and technical details were permitted.  The lack of complexity with DDoS attacks was also seen as a negative factor within Community B – the attacks were often considered too simplistic to be worth discussing.

The second innovation rejected by Community B was that of crypters, or crypting. The primary reason for this active rejection was a lack of complexity – in general, members believed that malware should be written in such a way that crypters are not necessary, with more-complex malware defeating virus detection and less-complex malware requiring crypters.  A secondary factor of image was also observed; some postings regarding crypters, if they were made by obvious script kiddies and were purely for malicious purposes with no interesting technical details, were also moved to the Board of Shame for the purpose of reduction of image of the member.

The last innovation that was rejected was SQL Injection within Community C, another active rejection.  Unlike the prior two innovations rejected within Community B, Community C appeared to reject SQL injections because they were not compatible with the needs of the community – namely, writing malware.  SQL injection is much more compatible with the needs of other communities, and was adopted by Communities A and D under the specific banner of the Havij SQL injection tool, and by Community B as a more general technique.  Interestingly, in many of these communities, this technique was associated with a severe reduction in image for adopters; while this did dissuade individual members from adopting the technique, it did not stop the general community from adopting.

Based upon the above discussion, it appears that the following adoption factors are most important when determining if an innovation will be rejected: image, complexity, and compatibility. A hacking innovation must first be compatible with the needs of the adopting community before it could be adopted; while the majority of proposed innovations were compatible with the communities where they were adopted, SQL injection was not compatible with Community C – a community focused upon writing malware, not upon executing attacks – and was rejected. This appears to be straightforward, but the next two factors are less obvious.

Interestingly, when considering complexity, on many occasions increased complexity was seen as a positive factor for adoption, and reduced complexity a negative factor. Agarwal and Prasad (1997) noted that increased complexity was not a barrier to adoption, as motivated adopters can overcome complex innovations if the innovation is important enough. Far from being a barrier, within the present study, complexity was often a requirement. If an innovation was seen as too simplistic or easy, it was not interesting enough for adoption, which seems to agree with the notion by Holt et al. (2012) that hackers are constantly trying to demonstrate their technical skills and abilities in order to show their proficiency. Crypters were rejected by Community B because they encouraged the development of code that was not complex enough to defeat antivirus, and were associated with "script kiddies", a term used to reduce the image of a member. DDoS attacks were rejected by Community B in part because of the lack of complexity in their execution. In other communities, complexity was also seen as important – within

Community C, many innovations were adopted because they were highly complex, and practices associated with a reduction in complexity, such as writing software in high-level, less powerful languages such as Visual Basic or .NET, were disparaged. The complexity of these innovations affected how potential adopters were viewed with respect to their skill level. This suggests, therefore, that complexity within the hacker communities examined is actually closely related to image.

Above all other factors, image was observed to be the most often *discussed* by potential adopters of an innovation, but appeared to have limited effect. Rarely was adoption of an innovation associated with an increase in image; however, certain innovations resulted in a severe reduction in image, and therefore image was observed to have a significant overall negative impact. Image played a primary role in the rejection of DDoS attacks by Community B, and certain classes of innovations – namely SQL injection (including Havij), as well as DDoS attacks in other communities – suffered from a decrease in image post-adoption. It is interesting to note that even though many innovations were influenced by image, it was not ultimately a critical adoption factor, as SQL injections were adopted by Communities A, B, and D, for example, even though many members felt strongly about the image of those who did adopt the technique. In fact, the innovations associated with a reduction in image, such as Havij in Community A, may actually be adopted in secret by members, who publicly refute the technique while confirming their use, as indicated by the prevalence of quotes containing phrases such as “I hacked my first website with this, altho now i prefer manually”. One interesting observation was made about the different types of image observed in each community while reviewing case study data – to each community, an increase or reduction in image meant different things. Table 34 describes the types of image observed in these communities.

**Table 34: Image Types in Different Communities**

| Community | Type of Image |
|---|---|
| Community A | Hacking Ability |
| Community B | Technical Mastery, Novelty of Attack |
| Community C | Programming Ability |
| Community D | Speed and Success |

As described in the above table, each of the above communities valued different types of image, and reputation points were awarded in each for members who demonstrated these qualities. In Community A, the members there were concerned with their perceived hacking ability; they wanted to establish "street cred" as a legitimate hacker and did not want to be perceived as a script kiddie. In Community B, while script kiddies were also derided, members of that community sought recognition for technically-sophisticated attacks that were novel and interesting enough to be worthy of discussion. Members who could introduce novel techniques were prized, and members who regurgitated the techniques of others, or who posted comments highlighting their lack of technical mastery, were shamed. In Community C, programming ability was the primary image motivation, and programmers who had significant skills were praised – and Russian programmers were recognized to be among the best. Community D, which consisted of the lowest-skilled hackers encountered in this study, members were concerned only with conducting successful attacks quickly, wanted the easiest methods to do so, and were generally unconcerned with their overall image in terms of ability.

Of the innovation factors predicted in the literature, one type was not observed – voluntariness. Moore and Benbasat (1991) predicted that both image and voluntariness would be important in IS innovation diffusion. For the communities examined in this research, image was indeed very important, but voluntariness was for the most part not observed. Within some communities, certain innovations – such as DDoS within Community B – were highly discouraged, to the point of strong social shaming when members proposed them. In a way, this indicates that rejection of this innovation is *involuntary* by community members. However, there were no occasions where specific innovations were mandated by the community, so outside of this involuntary rejection voluntariness was not found to be an important factor.

A review of the findings of the innovation-decision processes described for the above communities and innovations will also prove useful. These findings are summarized in Figure 23 and described below.

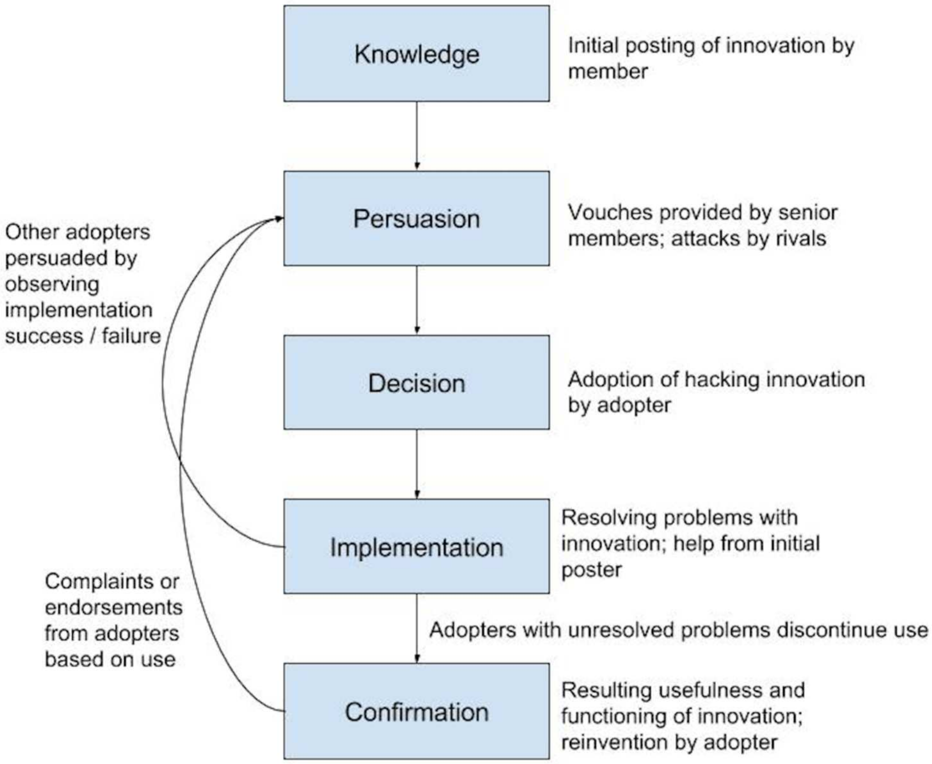

Knowledge
Initial posting of innovation by member
Persuasion
Vouches provided by senior members; attacks by rivals
Other adopters persuaded by observing implementation success / failure
Decision
Adoption of hacking innovation by adopter
Implementation
Resolving problems with innovation; help from initial poster
Complaints or endorsements from adopters based on use
Adopters with unresolved problems discontinue use
Confirmation
Resulting usefulness and functioning of innovation; reinvention by adopter

**Figure 23: Innovation-Decision Process within Hacker Communities**

It was noted that within communities that permitted vendors to sell tools and services, there were distinct differences between innovation-decisions involving innovations for sale and innovations that were offered for free. For innovations involving sales, vouches for vendors were found to be very important to the persuasion phase. Many of these vouches were provided by senior members of the communities, and there was an emphasis upon obtaining “vouch copies” of the hacking tools or services. These vouch copies permitted senior members to experiment with the innovation – where trialability and observability was noted to be important – and subsequently these members weighed in on the persuasion phase of the innovation-decision process. Strong vouches by senior members were found to be extremely important for sales threads, and when vouches were observed members tended to enter the decision phase and ultimately adopt.

Within the persuasion phase for sales threads, another trend was noticed: the public disparaging of a vendor by a rival. Within these threads, the rival vendor attempted to influence the persuasion phase with the goal of convincing members not to adopt. This was noted in the Amplification/DDoS threads in Communities A and C, where numerous vendors were offering DDoS services (often called booters or stressers) and these vendors were competing with each other for business. This was also observed with respect to crypters in Community C, where the sale of crypting tools was common. Typically, senior members or moderators would put a stop to these public spats, or in extreme cases would ban one member or another. Vouches by senior members also helped counteract these rivalries.

For those members who were persuaded and decided to adopt in the decision phase, the implementation phase was found to be critical. In this phase, adopters utilized the innovation and any problems were noted. Members who had problems described the technical issues they were having with the tool or technique, in an attempt to get support from the original poster and other community members. This phase was closely related with the persuasion phase – many members who were satisfied with their use of the innovation would vouch for the innovation in their own implementation phase, in an attempt to positively influence other members who might be in their own persuasion phase. Similarly, members who continued to have difficulties might attempt to convince members to avoid the innovation. If a user was not able to get their problem resolved during the implementation phase, many of them appeared to discontinue use in the confirmation phase. As a result, many of the members who posted the innovations were observed trying to publicly assist those who were describing problems, both so the use of the innovation was not discontinued by the member encountering issues, and so that other members who were deciding whether or not to adopt would not be dissuaded from adoption in their own persuasion and implementation phases.

When a member did make it through the implementation phase and entered the confirmation phase, the next decision was to continue use or to discontinue use. Many members posted messages indicating that they would continue to use the innovation, and similarly many members decided to discontinue use because the innovation ended up not being useful, or stopped working. In many cases, online tools offered by vendors – such as DDoS services – eventually stopped working as servers were suspended by hosting companies, leading customers to discontinue use and complain loudly. As in the implementation phase, these postings often influenced the decision phase of other members by offering vouches or complaints.

In some cases, the clearest sign of continued adoption was the reinvention of an innovation by an adopter. Examples of reinvention were observed in Communities A, B, and C, but not in D, where the volume of innovation activity in general was quite low. In Community A, the amplification DDoS technique was enthusiastically adopted and vendors immediately reinvented the innovation by incorporating it into their own booters and stressers, for a profit. In Community B, publicly-released crypters were adopted and improved upon by adding new features and then re-released by members, and SQL injection techniques were adopted and incorporated into standalone programs that were developed by members. And in Community C, the DDoS tool known as "minidos" was reinvented into a great number of other tools released there, and a number of crypters were also adopted, modified, and re-released, with the emphasis by the releasing members upon "crediting" the original author. The practice of writing tutorials was also a form of reinvention – the tutorial authors adopted a technique and developed new content to explain the technique to other members, primarily to cement their standing as a knowledgeable hacker within the community.

The importance of change agents within these communities also needs to be addressed. Three types of change agents were observed in this research: the original poster of an innovation, be it the actual developer or simply the member who introduces the tool or technique to the community; the senior member, who is able to use his significant influence to affect adoption; and the tutorial writer, who seeks to convince others to adopt the innovation. The overall influence of the original poster's innovation depended highly upon his seniority within the community, with the willingness of the community to experiment and ultimately adopt an innovation increasing with the poster's seniority level.  While some instances were noted where very junior members received enthusiastic responses, these were rare.  More-junior members were able to show their trustworthiness or skill by enlisting other members to vouch for them or to provide them reputation points, more of each which helped convince potential adopters to experiment with the innovation.

This leads to the next class of change agent – the senior member.  Senior members functioned as change agents primarily as gatekeepers through which innovations must pass.  These members were able to obtain "vouch copies" of tools or services so they could evaluate the innovations and offer their support.  The support of a senior member, through vouches or other positive feedback, was found to be a key component in adoption in most of the communities examined – in many cases, innovations which were unsupported or were negatively reviewed were not adopted, and innovations which were vouched for by senior members were adopted.  This phenomenon was observed in all four communities, to varying degrees – interestingly, even though one of the interviewees of Community B claimed that their community was egalitarian, it was Community C where the effect of seniority was observed to be the least.

The last class of change agents consisted of the tutorial authors.  These members, as explained above, sought to convince other members to utilize a given technique or tool by providing detailed instructions on how to operate the innovation.  This appears to have been primarily to demonstrate their own usefulness and skill to the rest of the community, or to gain reputation points.  Many tutorials were quite comprehensive and within certain communities emerged as the authoritative, go-to source for learning about the innovation, such as a SQL injection tutorial written by a Community B administrator that had been examined. Despite their apparent motivation of self-promotion, it appears that these tutorial authors were quite effective in enlisting other members to experiment with and adopt innovations, judging by the feedback provided by members who followed the tutorial.

The development of the innovation was stated by Rogers (2003) to be an important part of the innovation process. Recall that development, as defined by Rogers, consists of "the process of putting a new idea in a form that is expected to meet the needs of an audience of potential adopters" (Rogers, 2003, p.146). In the context of this study, development was observed to include such activities as the authoring of code around a particular technique, introduction of hacking services, or the posting of new hacking tools. These innovations may not have been "developed" by the individual member in a software engineering sense, but have been from a DOI perspective. Many of the innovations examined were developed outside of the community and introduced by change agents, where other members – both junior and senior – frequently experimented with, adopted, reinvented, and improved the innovations. The efforts were observed to be somewhat collaborative – usually the innovation would be introduced by the change agent in a complete form, but then others may comment or offer suggestions for improvement, which may or may not be incorporated by the initial developer. There were, however, some innovations which were developed within these communities. In Community A, this included amplification attacks – notably, Member A2 introduced the technique to the community, and he reportedly used that as a way to make significant amounts of money from selling tools and services related to it. In Community B, member-developed SQL injection scripts and crypters were noted. And in Community C, DDoS programs – notably “minidos”, which was reinvented in a number of other DDoS tools – as well as crypters were developed by members. In Community D, no significant development activity was observed, in accordance with the low skill level of the community.

# Chapter 5

# Conclusions

## 5.1 Introduction

This chapter concludes the research study by conducting a detailed discussion of the findings of this research in light of the original research questions and propositions. These findings include a review of the results from the four case study locations as well as those obtained from the cross-case analysis. A discussion of the importance and significance of the results is conducted. Practical implications for information security are also discussed, as are potential limitations of the research, an evaluation of the rigor of the research, and recommendations for future research. Finally, a conclusion section summarizes the research.

## 5.2 Findings

The original research model for this study, described in Figure 2, can be revisited with the insights obtained from the case studies and cross-case analysis. Table 35 details a summary of findings in relation to the original research model, which will guide the discussion in this section.

**Table 35: Relation of Findings to Research Model**

| **Innovation Stage** | **Key Findings** |
|---|---|
| Identification of Hacking Need | Need arises when existing hacking methods fail to be effective |
| Development of Innovation | Collaborative development within hacker community, combined with introduction of externally-developed innovations |
| Communication within Community | Primarily mass media, localite channels during knowledge phase; some interpersonal, localite channels during persuasion phase |
| Innovation Adoption Factors | Complexity: more complex hacking innovations favored and adopted; Image: can result in social pressures to not adopt hacking innovation, and different hacking communities value different types of image; Compatibility: innovation must be compatible with the needs of the hacking community |
| Adoption of Innovation | Most innovations adopted by hacking communities. Vouches by senior members important contributor to decision to adopt. |
| Use of Innovation | Implementation phase critical: problems with hacking innovation need to be resolved, or innovation is discontinued |
| Continued Use | Reinvention of hacking innovation a reliable signal of continued use |

Based on the above case study and cross-case analysis, and summary of findings in relation to the original research model, the original research questions of the study can be answered. Recall that both of the two initial research questions proposed in Chapter 1 were broken down into two further propositions. Rival propositions were also developed for each proposition to ensure that the behavior observed was explained by the proposition and not by the direct rival, as per Yin (2009). This section will examine these propositions and the rival propositions to determine if they were supported by the results of the case study.

### *5.2.1 Research Question One*

The first research question of this study was broken down into two propositions. These propositions, and their rivals, are provided and discussed in light of the results from Chapter 4.

*P1: Participants of hacking communities develop new techniques when existing hacking techniques cease to reliably breach targeted organizations.*

*R1: Participants of hacking communities develop new techniques as a means of demonstrating skill and earning status in the communities.*

The first proposition describes the hacking need that results in the introduction of the innovation to the community. A review of the hacking needs observed for each of the innovations examined in the four communities under study revealed support for this proposition in most cases. For example, the amplification technique was introduced when existing methods of DDoS ceased to be as effective. With increasing bandwidth available to victim servers, and the proliferation of anti-DDoS countermeasures available to organizations, standard methods of DDoS – such as sending data floods from a single server, or employing bots to send data from multiple computers – were becoming less effective. Amplification (and the sub-technique of reflection) made DDoS attacks many times more powerful than had been possible before, and as a result many vendors in these hacking communities offered powerful, and cheap, DDoS services for rent.

Similarly, crypters were introduced in response to a clearly defined need. Antivirus companies began implementing heuristic-based detection mechanisms, and malware that previously did not match known signatures was being caught by these IS security countermeasures, resulting in malware as an overall technique becoming less effective. Crypters were developed and released to these communities as a way to overcome the antivirus countermeasure employed by organizations. Another example can be found in bulletproof hosting – as web-based botnets became more popular, as a countermeasure targeted organizations and security companies would request (or legally compel) the providers hosting the malicious content to take down the malware. To allow these new malware techniques to continue to be effective, bulletproof hosting was needed so that malware would stay online.

In one case, a tool was introduced to these communities to make it easier for hackers to execute their attacks. SQL injection was developed as an overall technique as a new way to breach web-based assets of organizations – countering more traditional countermeasures such as firewalls, in line with P1 – but the tool Havij specifically was developed to allow attackers of all skill levels to engage in this technique. This democratization of the SQL injection technique allowed many more victim organizations to be breached.

Shodan was the only instance of a tool or technique being introduced without a specific hacking need – or at least, an observable need in the community where it was introduced, although it may have been created in response to the needs of other, unreviewed communities. When Shodan was introduced to Community A, members were interested and enthusiastic, but it did not appear that they were clamoring for a solution to find and exploit internet-connected devices, such as webcams, prior to the introduction of the tool. In this one instance, it appeared that the member who posted the innovation did make his post to demonstrate his skill and increase his own status (measured in terms of reputation points) within that Community. This was the only case where this was observed, so it appears that for the other innovations P1 is supported and R1 rejected. This indicates that these hacking communities primarily follow a "need-pull" model of technology innovation (Baskerville & Pries-Heke, 2001).

The second proposition for the first research question, and its corresponding rival, is below:

*P2: The methods by which the new hacking techniques are communicated amongst the community affect the adoption of the innovations.*

*R2: A hacking innovation, if useful enough, will be adopted by the community regardless of the manner in which it is communicated through the social system.*

The communications channels employed by community members when considering the examined innovations remained relatively consistent across the examined communities. The main communications channels employed by all four communities were the forum public messages themselves – which would be considered mass media, localite communications per Rogers (2003), with some conversations occurring over private messages, which would be considered interpersonal localite communications. Discussions over external chat services, such as ICQ, Jabber, or MSN, were also fairly common, and one community – Community B – operated its own official IRC server, where members congregated. The external chat services would be considered interpersonal, cosmopolite communications methods per Rogers, while the Community B IRC server would be considered a interpersonal localite method. In some cases, members also visited external websites, primarily to interact with vendors who may be operating services, such as DDoS booters or stressers, outside of the community, which would be mass media, but cosmopolite methods. These communications were overwhelmingly informal, in line with the overall informal, slang-heavy nature of hacker communications described by Holt (2009).

Rogers (2003) stated that cosmopolite, mass media channels are more often employed during the knowledge phase of the innovation-decision process, and localite, interpersonal methods are more often employed during the persuasion phase. This study suffers a bit from observation limitations, as most of the innovations were primarily observed over the public forum postings and insights into private messages were not possible. Similarly, most of the observed communications were localite and cosmopolite channels were referred to but generally not observed directly. However, on many occasions during the persuasion phase members would ask to be contacted over private message or external chat service, lending some support to Rogers' notion.

Many of the sales of tools and services also occurred over these interpersonal channels. Cosmopolite channels did not appear to be utilized as much as localite channels; however, some vendors offered external websites which contained information about their services, and some members provided their external chat handles to negotiate sales.

In each of the cases above where an interpersonal channel was employed, the community adopted the innovation. Generally these communications involved the original poster of the message – the vendor, or the change agent who is seeking adoption of the innovation – being contacted by a potential adopter. It seems that this "personal touch" does help with the adoption, although for interpersonal communications no distinction was made between cosmopolite and localite methods. This would imply support for P2, but does not completely refute R2, suggesting that further research encompassing cosmopolite channels would help to answer this question.

Now that the first two propositions have been discussed, a discussion of the first research question can be conducted. This question is reproduced below:

> *RQ1: What characteristics affect the development and communication of emergent hacking techniques and tools within hacker communities?*

We have learned that hacker tool and technique innovations are developed and introduced into communities in response to a pre-defined hacking need because existing methods of conducting attacks are inadequate, in most instances (with the exception of Shodan). This is in line with Utterback's (1971) findings of innovation introduction within organizations. We have also learned that most of the communications channels within these communities tend to be localite mass media channels, and that interpersonal channels (of either the localite or cosmopolite variety) have a positive impact upon the eventual adoption of the innovation.

The development of the innovation within these hacking communities, as defined by Rogers (2003), includes both software-engineering style development of code, as well as the introduction of hacking tools and services to the communities. Many of the innovations observed during this research were written outside of the individual communities, while some of them were in fact authored by community members for the use of the individual community. This development was observed to be collaborative in nature, with community members commenting upon code changes and feature improvements. These innovations were introduced in the knowledge phase of the innovation-decision process and later championed by a variety of change agents, who sought to influence adoption within the community.

### *5.2.2 Research Question Two*

The second research question of this study was also broken down into two propositions. The first proposition and its corresponding rival is described below:

> *P3: Change agents within the hacking community are instrumental in the adoption of the hacking innovation.*
>
> *R3: The role of change agents within a community advocating for particular innovations is minimal; instead, the merit of the innovation dictates its adoption.*

Change agents were found to be quite influential within the studied communities, as described in detail in the previous chapter. Within these communities, there were three types of change agents observed, each of which could potentially overlap – the member who introduced the innovation, the senior member of the community, and the tutorial author. For members who introduced various hacking tools or techniques, the seniority of the member was found to be quite important, with innovations introduced by more-senior members receiving wider interest, experimentation, and eventual adoption of their innovations, and innovations introduced by more-junior members experiencing less success in adoption. These more junior members would often need to enlist the support of other members through vouches or reputation points before their innovation was considered for adoption.

Senior members themselves were a class of change agent, by acting as gatekeepers through which hacking innovations needed to pass. Senior members in the studied communities fulfilled Holt's (2013) observation of a "relational J curve" structure in hacking communities, where a small number of users – the senior users – are ultimately responsible for a majority of the useful content. Innovations that received the support of these members – which included site administrators, moderators, or long-time members with high numbers of posts – were much more likely to be adopted by the community. Negative reviews by senior members often had a severe negative impact upon the adoption of hacking innovations, as did a simple lack of support. Finally, tutorial authors were found to be the third type of change agent. These members often wrote tutorials to demonstrate their skill with a particular innovation, and innovations which had tutorials written about them appeared to be adopted more quickly and enthusiastically.

Based upon the above discussion of change agents, it appears that P3 is supported. R3 likewise appears to be rejected, based upon the importance of senior members and influence of tutorial authors. Despite the insistence by some interviewees that their communities are indeed egalitarian and are merit-based, tool innovations which did not receive the support of senior members, or received a negative review, were often not adopted.

The final proposition considered in this research, P4, can be found below along with its corresponding rival:

> *P4: Continued use of a hacking innovation by community members will determine the ultimate adoption of an innovation.*
>
> *R4: Ultimate adoption of an innovation is unaffected by continued use of the hacking innovation.*

The case study conducted on these four communities revealed that continued use of an innovation was indeed a predictor of overall adoption. Many tool sub-innovations were experimented with but members struggled in the implementation phase, leading the members to eventually discontinue use in the confirmation phase and for the individual tools to not be adopted, even if the overall technique was eventually adopted by the larger community. This provides support for the rejection of rival R4. On the other hand, when members resolved issues in the implementation phase (often with the assistance of the original tool author), those innovations appeared to have been adopted at a higher rate. Tool and technique reinvention was also found to have a strong association with adoption. When members reinvented tools or techniques, such as amplification attacks in Community A or minidos in Community C, or authored significant amount of tutorials (themselves a form of reinvention) regarding tools or techniques, these innovations were among the most widely adopted of those reviewed. This indicates that P4 should be accepted. As described by Parthasarathy and Bhattacherjee (1998), this post-adoption behavior was ultimately important in understanding the success of these innovations.

Now that propositions P3 and P4 have been discussed, an analysis of RQ2 is appropriate. Recall that RQ2 was as follows:

> *RQ2: How do the participants in these communities decide to adopt these techniques and employ them in attacks against organizations?*

To answer this question, the importance of change agents, from proposition P3, should be considered. However, also important were innovation diffusion factors as well as the various phases of the innovation-decision process relevant to this aspect of adoption.

While all of the innovation factors discussed played some particular role – with the exception of voluntariness, which was not relevant for these communities – certain factors were determined to be more important. Complexity was found to be important, but for counterintuitive reasons – it turned out that increased complexity was beneficial to adoption, and a lack of complexity was seen as a negative factor. Innovations which were seen as too simplistic were disparaged by members and were often associated with "script kiddies", which frequently – but not always – led to non-adoption. For example, in Community B DDoS attacks were seen as too simplistic to be worth discussing, and they were not adopted. Similarly, usage of crypters in Community B were seen as indicative of low programming skills, as highly skilled developers should be able to make their malware undetectable without the use of a third-party crypter, leading to the rejection of the technique by the community. This indicates that complexity is closely tied to image, as the complexity of an innovation affects the image of the adopter.

Image itself was found to be somewhat important – in some communities, the negative image associated with an innovation (such as DDoS in Community B) contributed to the community's rejection of the technique. However, in other communities certain tools or techniques, such as the Havij tool or SQL injection more broadly, were adopted even though usage of them was often associated with "script kiddies". One interesting finding regarding image was that different communities valued different types of image: in Community A, overall hacking ability was prized; in Community B, technical mastery and novelty were rewarded; in Community C, programming ability was what was recognized; and in Community D, the speed of attacks and the end results were important, with little concern how a hacker achieved them.

Compatibility was the last innovation factor determined to be critical for adoption. When an innovation was deemed to be compatible with the needs of a community, it was often adopted. However, when the innovation was determined to be incompatible with the needs of the community – such as SQL injection in Community C, which was generally not compatible with the malware-writing needs of the community – it was not adopted. For the most part, the innovations examined in this study were compatible with the needs and goals of their communities.

Regarding the innovation-decision process, while all of the phases played a role in the eventual adoption or rejection of an innovation, a few of the phases of the process appeared to be especially crucial within these hacker communities. For example, the persuasion phase was observed to be very active, and for innovations associated with vendors who were trying to make a profit from the innovation, this phase was quite important. These vendors frequently sought vouches from senior members who could support the innovation, and often provided “vouch copies” for free to senior members in order to obtain support. After vouches were provided by these senior members, many of the more-junior members would declare their intention to adopt. Also observed in the persuasion phase for these sales threads were “anti-vouches”, or disparaging remarks from rival vendors, who attempted to convince potential adopters to not adopt the innovation, often in favor of their own.

The implementation phase was also observed to be critical, as detailed in the response to proposition P4. In this phase, adopters described their experiences with the innovation, and often publicly detailed their problems in an attempt to get them resolved. If these members were able to get their problems with the innovation resolved, they appeared to continue use of the innovation; likewise, if they were not able to resolve the problems, they appeared to discontinue use. The change agents championing the innovation often attempted to publicly assist these adopters with their problems. This behavior was also closely related to the persuasion phase, as members who were considering adoption observed these exchanges and could decide to adopt or reject based on the implementation experiences of other members.

In the confirmation phase, also related to proposition P4, one of the best indicators that an innovation was going to be adopted by the community was when individual members would reinvent the innovation in their own offerings, in line with what had been predicted by Rogers (2003). This reinvention could take different forms, including the repackaging of techniques (such as amplification attacks) by vendors into their own for-profit offerings, or code from tools (such as minidos) that were incorporated into new tools, often with the author of the new tool crediting the original source. Tutorials, frequently observed in these communities and whose presence often signaled that a community had adopted a particular tool or technique, were also a form of reinvention; tutorial authors wrote new content around an innovation and released that content to the community to aid in the adoption of the innovation.

The above innovation factors and important phases from the innovation-decision process, combined with our discussion of the importance of change agents, greatly aids in our understanding of the adoption process of these hacking innovations and in the second research question of the study.

### 5.3 Discussion

This study attempted to understand how hacker communities innovate to overcome IS security countermeasures. There are a number of important findings from this research, which significantly add to the body of knowledge of IS security literature as well as the technology adoption and innovation literature. While studies of IS security adoption and innovation within organizations are plentiful, this is the only known study which examines the innovation and adoption activities of blackhat hackers, in the underground communities where they spend much of their time.  These hackers are the human agents behind the attacks that cause organizations to spend so much money, and develop new countermeasure technologies, so detailed study of their own innovation processes was found to be lacking in the literature.  In fact, Mahmood et al. (2010) argue that more research focused on the activities of these blackhats is necessary, even though it may be harder for researchers to conduct.  Studies of criminal innovations themselves are fairly rare (Baker & Faulkner, 2003; Koller, 2010; Snynder, Priem, & Levitas, 2009), and to date, none have been observed which focus upon criminal or malicious IS innovations. Given the pace and breadth of hacking innovations observed in these communities, there are additional opportunities for study which may further enhance the IS security body of knowledge.

In the arena of innovation and adoption literature specifically, this study uncovered some counterintuitive findings.  While this research found that hacker communities reject innovations that are not compatible with the needs of the community – a seemingly-obvious conclusion – it also found that these communities reject innovations because of a lack complexity or the negative image associated with adopting a particular hacking innovation.  Agarwal and Prasad (1997) noted that increased complexity is not always a barrier to adoption, as motivated adopters will overcome complex challenges; however, the present study has demonstrated that hackers seek out highly complex innovations and shun overly simplistic ones, which makes hacking communities unique in the innovation literature.  This study also highlighted the strong social pressures exerted through the threat of negative image within some communities to not adopt certain innovations, or risk being labeled as a “script kiddie” or having your post displayed on the “Board of Shame”. Prior innovation research has focused primarily upon the increase in image associated with adopting a particular innovation, not the reduction in image, suggesting a new arena of study for innovation researchers.

This study also confirmed the importance of hacking communities as a separate subculture where hackers learn from each other and seek to gain status, as described in numerous studies of hacker communities (Bratus, 2007; Holt & Copes, 2010; Jordan & Taylor, 1998). Additionally, many of the activities that occurred within these communities – including posting new innovations, vouching for the activities of others, or providing feedback on new tools and techniques – were done by hacker participants who sought to gain “rep” and an elevation of image. This importance of image also may have practical implications, as described in the next section.

### 5.4 Practical Implications

There are a number of implications for policy and practice from this research. By identifying the importance of change agents in the hacker tool and technique innovation process, parties seeking to disrupt this process – such as law enforcement, information security firms, or even targeted organizations – know which hackers to target to achieve maximum impact. This disruption could include arresting senior members to remove their ability to support innovations, or coopting them to provide negative vouches to emerging innovations. Tutorial authors could also be targeted in a similar fashion. Furthermore, by understanding which innovation factors were found to have the most impact – image, complexity, and compatibility – organizations could seek to engage in online undercover operations within these communities to influence the innovation-decision process.  While Holt (2013) has argued that engaging in simple slander attacks against hacker community members may be ineffective, this study may indicate that more sophisticated manipulations that focus upon these factors – such as complaints about the functionality of innovations, labeling innovations as too simplistic and more appropriate for script kiddies, or declaring certain classes of innovations, such as DDoS, not compatible with the objectives of a particular community – may have greater effect.

Another practical avenue may involve the economic disruption of the marketplaces observed in many of the communities in this study.  Many of the postings in these forums were financial in nature – members were selling tools or services for individual profit, and this marketplace, and the importance of reputation for these sellers, seemed to drive much of the innovation activity observed.  A focus upon disrupting these transactions, by introducing uncertainty into negotiations or by physically blocking them from occurring by targeting payment platforms, may make it harder for hacking tool or service sellers to profit from their innovations and may lead to a reduction in innovation activity overall.

Organizations could also seek to develop and utilize new or improved IS security countermeasures which incorporate the findings of this study.  For example, hackers were found to discourage the use of overly simplistic innovations, favoring more complex innovations, due in part to the reduction of image associated with the simpler innovations.  By focusing energies on defending against the more-complex innovations, organizations may sow confusion within the communities if the complex innovations cease to be as effective, by pitting the desire for increased complexity against the need for success.

### 5.5 Assumptions, Limitations, and Delimitations

This research included a number of assumptions, limitations, and delimitations. One main assumption was that the participants of the hacking communities under study were actually engaging in the attacks that they claim to be, and as such are representative of hackers in general. It is possible that some participants may not be conducting the attacks they claim, or may even be undercover law enforcement agents or other researchers; however, this research has taken the participants, their stated abilities, and their claims of past exploits at face value. It was also assumed that interview participants were able to answer questions honestly; to help ensure this, the identities of the participants were kept anonymous and confidential.

The preset study also had several limitations. For example, this study described innovation activities within a number of hacking communities, but these findings may not be representative for other hacking communities that may vary in size, membership, language, or other factors. Another limitation of the study is that of time; this study occurred over a certain period of time, and as such reflected the activities, attitudes, and other conditions of the hacking communities during that time, which may not necessarily reflect what will occur at a later date.

To control the scope of this research endeavor, a number of delimitations were imposed. For example, only those communities that primarily utilize the English language were examined. Also, only those communities to which the researcher was able to obtain access were investigated, such as public registration for web forums. To further control the scope, only four cases were be analyzed.

### 5.6 Evaluation of Research Rigor

As described in earlier chapters, this study applied a positivist philosophical approach to the understanding of hacker innovation. To evaluate the rigor of this study, in addition to the previous discussion of threats to validity described in Chapter 3, the evaluation criteria proposed by Dubé and Paré (2003), which concerns positivist case studies specifically, will be employed.

Dubé and Paré (2003) lay out a number of criteria for the design of explanatory case studies. First, they should have *clear research questions*, focusing upon well-defined and clearly stated “how”, “why”, or “what” questions. The *unit of analysis* should also be specifically stated. They should also employ a *theory of interest* and derive predictions from that theory, and should also include rival propositions to account for alternative explanations. A successful and rigorous case study will also employ *multiple-case design* and replication logic within those cases. The *case context* should be provided, consisting of a site description of the case and the time period of the case. All of these recommendations for research design were followed in this study. One design recommendation which was not followed was the employment of a *team of researchers* to collect data, as only a single researcher was utilized.

Similar evaluation criteria were provided by Dubé and Paré (2003) for the collection of data within explanatory case studies. The *data collection process* should be elucidated, so readers of the research can determine which methods were employed. *Multiple methods* of collection should be utilized to achieve *triangulation* of the data. In the present study, the multiple methods of data collection were clearly described and triangulation of data was achieved. A *mix of qualitative and quantitative data* should be collected; in the present study, most of the collected data was qualitative, but some quantitative data – namely, the frequency of conversations about specific innovations – was collected. A *case study protocol* should be employed, and a *case study database* constructed to house the data – both of which were employed by the present study.

Finally, Dubé and Paré (2003) detail evaluation criteria for data analysis within explanatory, positivist case studies. These include employing *field notes* and *coding*, as well as *data displays* such as tables, all of which were employed in the present study. A *logical chain of evidence* should be maintained so a reader can follow the conclusions draw, and analytic strategies such as *pattern-matching*, *explanation-building*, and *time-series* should be employed. The present study indeed maintained a chain of evidence, and employed a combination of pattern-matching and explanation-building approaches. A time-series approach was not employed because the present study did not have an explicit time component, and a key feature of time-series analysis often involves the introduction of a particular intervention or improvement (Runeson & Höst, 2009), which was not a feature of this research. *Cross-case analysis* should also be utilized to draw conclusions across cases, which occurred in this research. Finally, using *quotes* from research subjects allows the reader to come to the same independent conclusions as the researcher, and were used extensively in this study.

It should be noted that Straub, Boudreau, and Gefan (2004) also provided criteria for evaluating rigor in positivist research. However, their focus was on quantitative techniques, with very few mentions of qualitative methods. Some of the criteria they specify which could apply to qualitative research includes *inter-rater reliability*, which does not apply to this study as only a single researcher performed data collection, and *content validity* of the instrument through a literature review and expert panel.  In the present study, a literature review did guide the creation of the instrument (case study protocol); however, the final protocol was not reviewed by the expert panel convened earlier in the research.  This could be addressed in future research studies on this topic.

## 5.7 Recommendations for Future Research

This study has created a number of opportunities for future research.  For example, the unit of analysis of this study was the hacker community itself, which resulted in a focus upon community-wide trends and behaviors, but with less focus upon aspects of DOI that pertain to individuals and their influences.  These aspects include classifying individual adopters into adoption categories – such as early adopters, laggards, and so forth – and quantifying how many of each category were present for various innovations, as well as what critical mass of adopters much be present for the community as a whole to adopt.  Future research could examine the impact of individual hackers within the community and follow these individuals as they interact with various innovations, to determine which ones they support or reject, and why.  Such research could also examine individual hackers to determine if they might be placed into different adoption categories for different types of innovations.

Future research could also explore in greater depth communications that occur outside of the communities, to answer questions about the efficacy of different communication types and locations.  In addition, subsequent research endeavors could examine these communities from a cross-cultural perspective, to determine if hackers from different cultures or who speak different languages adopt hacking tools and techniques in different fashions. One final area of future research may address the possibility that outwardly unpopular innovations – such as Havij – may in fact be adopted in secret to preserve the image of adopters.

## 5.8 Conclusion

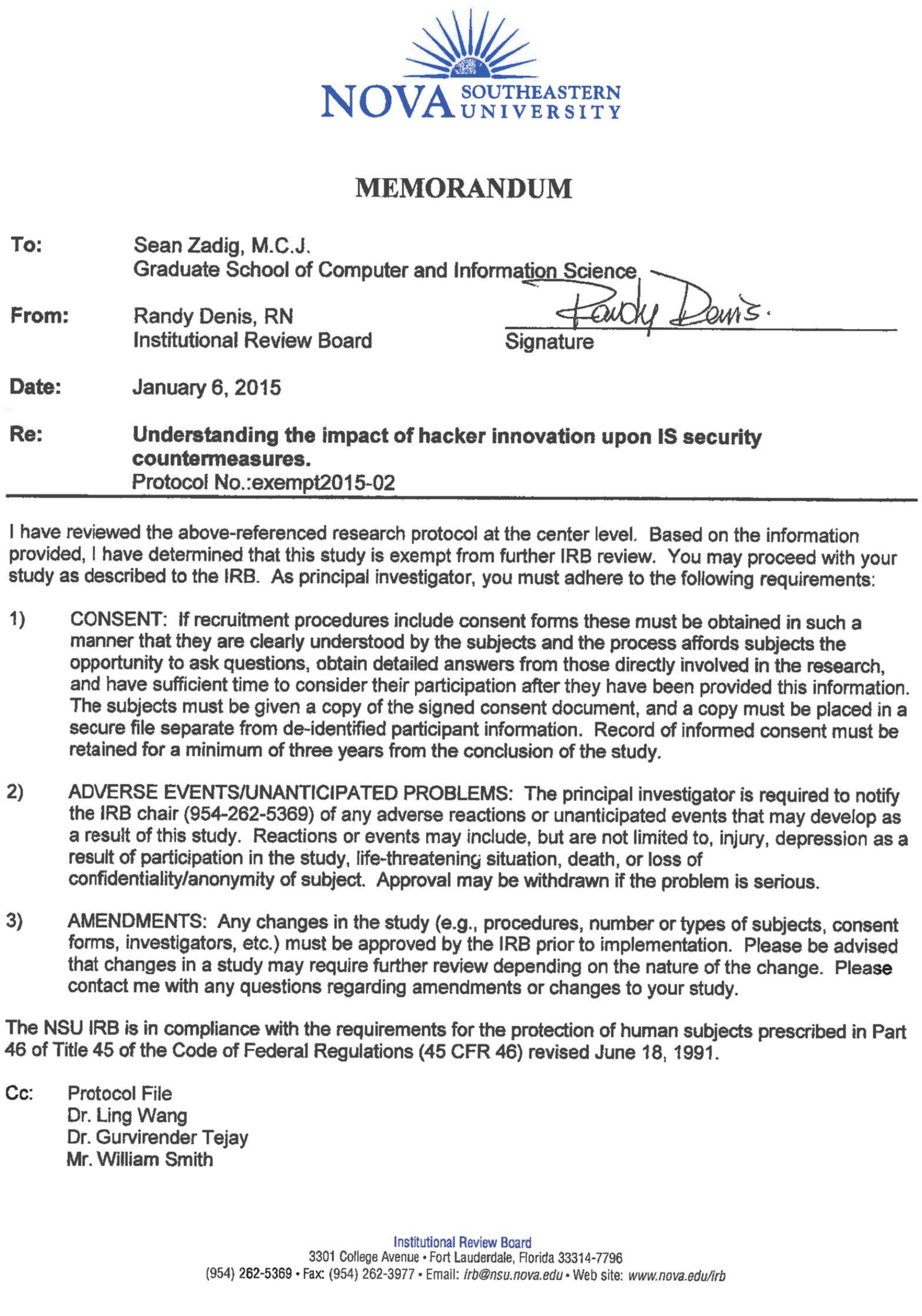
This research conducted a comprehensive multi-part case study of four distinct hacking communities, applying a Diffusion of Innovations lens to understand how these communities adopted or rejected hacking innovations that had diffused through the communities. These innovations were developed within these underground communities to overcome IS security countermeasures erected by organizations to defend against an ever-increasing barrage of cyberattacks. The cycle of innovation, attack, and countermeasure is incredibly costly for defending organizations, an equation that is wildly out of balance, with the defenders bearing the overwhelming brunt of costs. Further insight into the hacker innovation diffusion process within these communities was needed to begin to understand how to disrupt this cycle.

This research highlighted a number of key findings where future deterrent and disruption efforts may be targeted. First, hackers within these communities highly value innovations which fill a specific hacking need, often to overcome specific IS security countermeasures such as firewalls, antivirus, and DDoS protections. Change agents were also found to be very important – these change agents included the developer of the hacking tool or technique innovation, senior hackers within the underground hacking community, and the authors of hacking tutorials, each of whom had the ability to champion an innovation and influence its adoption. To be successfully adopted and diffused, these innovations that are introduced to and developed within these communities need to be compatible with the needs of the members and need to be complex enough so that hackers can adopt them without being perceived as having low skills. This requirement of complexity for perception's sake is related to the overall image that a hacking innovation confers on a potential adopter. Image by itself was found to be an important factor, but not a critical one, as many hacking innovations which conferred a lower image – such as SQL injection and Havij – were adopted, perhaps reluctantly, by these communities.

This research also examined the innovation-decision process that occurs when hacking innovations are evaluated and adopted by the hackers that make up the members of the communities in this study. The persuasion phase of the process was found to be quite important within these communities – senior members often vouched for innovations they found to be worthy, and members selling innovations sought these vouches by providing "vouch copies" of hacking tools or services – such as bulletproof hosting services, or DDoS tools – to senior members. Developers of rival hacking innovations, such as operators of competitor booter services, also frequently attempted to disparage innovations in this phase and convince members not to adopt. The implementation phase was also found to be critical, as during this phase members would frequently discuss problems they were having with the hacking tools and techniques they were in the process of adopting. If these problems were not resolved to the satisfaction of the adopter, the innovation was frequently rejected – and this occurred in public, so other members in their own persuasion phases were influenced by the outcome of the implementation phase of the hacking innovation. In contrast, innovations where the developer was able to resolve the issue of the adopter experienced increased adoption by other hackers. Finally, the confirmation phase was an excellent indicator of whether an innovation was discontinued or continued to be adopted. One of the most reliable indicators of continued use was the reinvention of the innovation, as other members would incorporate the innovation – either techniques, such as amplification attacks, or direct code, such as minidos – into their own tools that were released to the community. Tutorial writing was also found to be a form of reinvention that was popular within some communities, and was a reliable indicator that the community had successfully adopted the hacking innovation.

This study of the innovation diffusion and adoption process for the examined innovations was a starting point in the overall understanding of hacker innovations, but was illuminative in its own right. By incorporating the findings of this research, organizations and IS security firms can develop approaches to disrupt the innovation diffusion process and either develop appropriate defenses, or prevent these hacking tool and technique innovations from reaching their IS security countermeasures in the first place.

# Appendix A: Institutional Review Board Approval



NOVA SOUTHEASTERN UNIVERSITY

## MEMORANDUM

**To:** Sean Zadig, M.C.J.
Graduate School of Computer and Information Science

**From:** Randy Denis, RN
Institutional Review Board

Randy Denis.
Signature

**Date:** January 6, 2015

**Re:** **Understanding the impact of hacker innovation upon IS security countermeasures.**
Protocol No.:exempt2015-02

I have reviewed the above-referenced research protocol at the center level. Based on the information provided, I have determined that this study is exempt from further IRB review. You may proceed with your study as described to the IRB. As principal investigator, you must adhere to the following requirements:

1) CONSENT: If recruitment procedures include consent forms these must be obtained in such a manner that they are clearly understood by the subjects and the process affords subjects the opportunity to ask questions, obtain detailed answers from those directly involved in the research, and have sufficient time to consider their participation after they have been provided this information. The subjects must be given a copy of the signed consent document, and a copy must be placed in a secure file separate from de-identified participant information. Record of informed consent must be retained for a minimum of three years from the conclusion of the study.

2) ADVERSE EVENTS/UNANTICIPATED PROBLEMS: The principal investigator is required to notify the IRB chair (954-262-5369) of any adverse reactions or unanticipated events that may develop as a result of this study. Reactions or events may include, but are not limited to, injury, depression as a result of participation in the study, life-threatening situation, death, or loss of confidentiality/anonymity of subject. Approval may be withdrawn if the problem is serious.

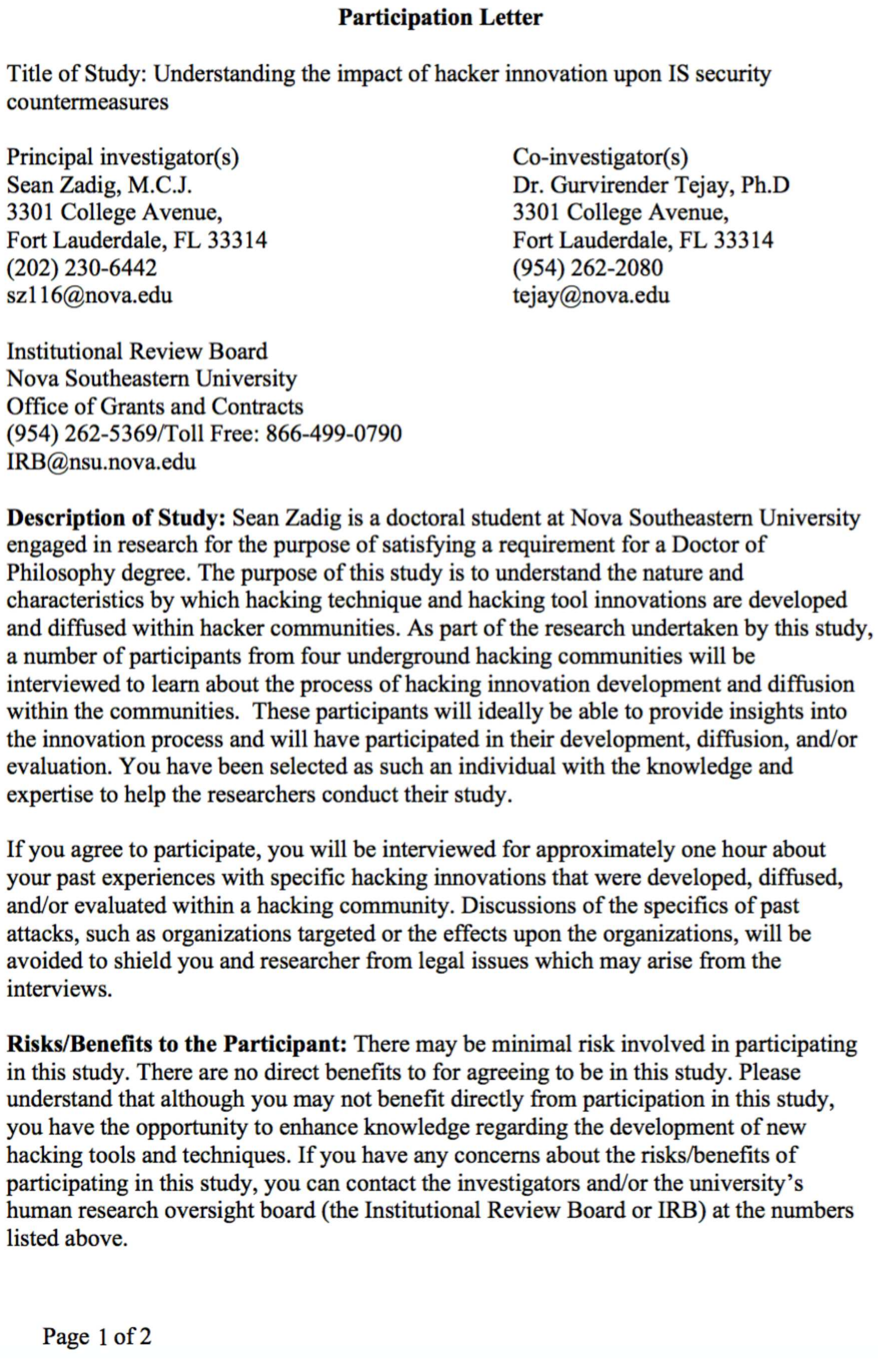
3) AMENDMENTS: Any changes in the study (e.g., procedures, number or types of subjects, consent forms, investigators, etc.) must be approved by the IRB prior to implementation. Please be advised that changes in a study may require further review depending on the nature of the change. Please contact me with any questions regarding amendments or changes to your study.

The NSU IRB is in compliance with the requirements for the protection of human subjects prescribed in Part 46 of Title 45 of the Code of Federal Regulations (45 CFR 46) revised June 18, 1991.

Cc: Protocol File
Dr. Ling Wang
Dr. Gurvirender Tejay
Mr. William Smith

Institutional Review Board
3301 College Avenue • Fort Lauderdale, Florida 33314-7796
(954) 262-5369 • Fax: (954) 262-3977 • Email: *irb@nsu.nova.edu* • Web site: *www.nova.edu/irb*

## Appendix B: Interviewee Participation Letter



## Participation Letter

Title of Study: Understanding the impact of hacker innovation upon IS security countermeasures

Principal investigator(s)  
Sean Zadig, M.C.J.  
3301 College Avenue,  
Fort Lauderdale, FL 33314  
(202) 230-6442  
sz116@nova.edu

Co-investigator(s)  
Dr. Gurvirender Tejay, Ph.D  
3301 College Avenue,  
Fort Lauderdale, FL 33314  
(954) 262-2080  
tejay@nova.edu

Institutional Review Board  
Nova Southeastern University  
Office of Grants and Contracts  
(954) 262-5369/Toll Free: 866-499-0790  
IRB@nsu.nova.edu

**Description of Study:** Sean Zadig is a doctoral student at Nova Southeastern University engaged in research for the purpose of satisfying a requirement for a Doctor of Philosophy degree. The purpose of this study is to understand the nature and characteristics by which hacking technique and hacking tool innovations are developed and diffused within hacker communities. As part of the research undertaken by this study, a number of participants from four underground hacking communities will be interviewed to learn about the process of hacking innovation development and diffusion within the communities. These participants will ideally be able to provide insights into the innovation process and will have participated in their development, diffusion, and/or evaluation. You have been selected as such an individual with the knowledge and expertise to help the researchers conduct their study.

If you agree to participate, you will be interviewed for approximately one hour about your past experiences with specific hacking innovations that were developed, diffused, and/or evaluated within a hacking community. Discussions of the specifics of past attacks, such as organizations targeted or the effects upon the organizations, will be avoided to shield you and researcher from legal issues which may arise from the interviews.

**Risks/Benefits to the Participant:** There may be minimal risk involved in participating in this study. There are no direct benefits to for agreeing to be in this study. Please understand that although you may not benefit directly from participation in this study, you have the opportunity to enhance knowledge regarding the development of new hacking tools and techniques. If you have any concerns about the risks/benefits of participating in this study, you can contact the investigators and/or the university's human research oversight board (the Institutional Review Board or IRB) at the numbers listed above.

**Cost and Payments to the Participant:** There is no cost for participation in this study. Participation is completely voluntary and no payment will be provided.

**Confidentiality:** Information obtained in this study is strictly confidential unless disclosure is required by law. All data will be secured in a locked filing cabinet. Your name will not be used in the reporting of information in publications or conference presentations.

**Participant's Right to Withdraw from the Study:** You have the right to refuse to participate in this study and the right to withdraw from the study at any time without penalty.

**I have read this letter and I fully understand the contents of this document and voluntarily consent to participate. All of my questions concerning this research have been answered. If I have any questions in the future about this study they will be answered by the investigator listed above or his/her staff.**

**I understand that the completion of this interview implies my consent to participate in this study.**

## Reference List

Acquisti, A., Friedman, A., & Telang, R. (2006). Is there a cost to privacy breaches? An event study. *Procedings of the Fifth Annual Workshop on the Economics of Information Security*, Cambridge, England.

Agarwal, R. & Prasad, J. (1997). The role of innovation characteristics and perceived voluntariness in the acceptance of information technologies. *Decision Sciences*, *28*(3), 557-582.

Anagnostopoulos, M., Kambourakis, G., Kopanos, P., Louloudakis, G., & Gritzalis, S. (2013). DNS Amplification Attack Revisited. *Computers & Security, 39*(2), 475-485.

Anderson, J.M. (2003). Why we need a new definition of information security. *Computers & Security*, *22*(4), 308-313.

Anderson, R. (2001). Why information security is hard - an economic perspective. *Proceedings of the 17th Annual Computer Security Applications Conference*, 358-365.

Anderson, R., Barton, C., Bohme, R., Clayton, R., van Eeten, M.J.G., Levi, M., Moore, T., & Savage, S. (2012). Measuring the cost of cybercrime. *Proceedings of the 11th Annual Workshop on the Economics of Information Security*, Berlin, Germany.

Aubert, B.A., Rivard, S., & Patry, M. (2004). A transaction cost model of IT outsourcing. *Information & Management*, *41*(7), 921-932.

Bachmann, M. (2010). The risk propensity and rationality of computer hackers. *International Journal of Cyber Criminology*, *4*(1), 643-656.

Baker, W.E. & Faulkner, R.R. (2003). Diffusion of fraud: Intermediate economic crime and investor dynamics. *Criminology*, *41*(4), 1173-1206.

Bandara, W., Gable, G.G., Rosemann, M. (2005). Factors and measures of business process modelling: Model building through a multiple case study. *European Journal of Information Systems*, *14*, 347-360.

Barber, R. (2001a). Hackers profiled – who are they and what are their motivations? *Computer Fraud & Security*, *2*(1), 14-17.

Barber, R. (2001b). Hacking techniques: the tools that hackers use, and how they are evolving to become more sophisticated. *Computer Fraud & Security*, *3*(1), 9-12.

Baskerville, R. (1993). Information systems security design methods: Implications for information systems development. *ACM Computing Surveys*, *25*(4), 375-414.

Baskerville, R. & Pries-Heje, J. (2001). A multiple-theory analysis of a diffusion of information technology case. *Information Systems Journal*, *11*(3), 181-212.

Baskerville, R. & Siponen, M. (2002). An information security meta-policy for emergent organizations. *Logistics Information Management*, *15*(5), 337-346.

Bell, R.E. (2002) The prosecution of computer crime. *Journal of Financial Crime*, *9*(4), 308-325.

Benbasat, I., Goldstein, D.K., & Mead, M. (1987). The case research strategy in studies of information systems. *MIS Quarterly*, *11*(3), 369-386.

Biryukov, A., Pustogarov, I., & Weinmann, R. (2013). Trawling for tor hidden services: Detection, measurement, deanonymization. *2013 IEEE Symposium on Security and Privacy*, 80-94.

Bishop, M. (2000). Analysis of the ILOVEYOU worm. Retrieved February 23, 2013, from http://nob.cs.ucdavis.edu/classes/ecs155-2005-04/handouts/iloveyou.pdf.

Bishop, M. (2003). *Computer security: Art and science*. Boston, MA: Pearson Education.

Bleaken, D. (2010). Botwars: the fight against criminal cyber networks. *Computer Fraud & Security*, *2010*(5), 17-19.

Bodenheim, R., Butts, J., Dunlap, S., & Mullins, B. (2014). Evaluation of the ability of the Shodan search engine to identify Internet-facing industrial control devices. *International Journal of Critical Infrastructure Protection*, *7*(2), 114-123.

Boudreau, M., Gefan, D., & Straub, D.W. (2001). Validation in IS research: A state-of-the-art assessment. *MIS Quarterly*, *25*(1), 1-24.

Bradford, M. & Florin, J. (2003). Examining the role of innovation diffusion factors on the implementation success of enterprise resource planning systems. *International Journal of Accounting Information Systems*, 4, 205-225.

Broadhurst, R., Grabosky, P., Alazab, M., Bouhours, B., & Chon, S. (2014). An analysis of the nature of groups engaged in cyber crime. *International Journal of Cyber Criminology, 8*(1), 1-20.

Bratus, S. (2007). What hackers learn that the rest of us don’t: Notes on hacker curriculum. *IEEE Security & Privacy*, *5*(4), 72-75.

Buzzard, K. (1999). Computer security - What should you spend your money on. *Computers & Security, 18*(4), 322–334.

Campbell, K., Gordon, L.A., Loeb, M.P., & Zhou, L. (2003). The economic cost of publicly announced information security breaches: Empirical evidence from the stock market. *Journal of Computer Security*, *11*(3), 431-448.

Cavusoglu, H., Mishra, B., & Raghunathan, S. (2004a). The effect of internet security breach announcements on market value: Capital market reactions for breached firms and internet security developers. *International Journal of Electronic Commerce*, *9*(1), 69-104.

Cavusoglu, H., Mishra, B., & Raghunathan, S. (2004b). A model for evaluating IT security investments. *Communications of the ACM*, *47*(7), 87-92.

Cavusoglu, H., Mishra, B., & Raghunathan, S. (2005). The value of intrusion detection systems in information technology security architecture. *Information Systems Research*, *16*(1), 28-46.

Cavusoglu, H., Cavusoglu, H., & Raghunathan, S. (2004). Economics of IT security management: Four improvements to current security practices. *Communications of the Association for Information Systems*, *14*, 65-75.

Cheng, J.M., Kao, L.L., & Lin, J.Y.C. (2004). An investigation of the diffusion of online games in Taiwan: an application of Roger's diffusion of innovation theory. *The Journal of American Academy of Business*, *5*(1), 439-445.

Chetty, S. & Holm, D.B. (2000). Internationalisation of small to medium-sized manufacturing firms: a network approach. *International Business Review*, *9*(1), 77-93.

Chircu, A.M. & Kauffman, R.J. (2000). Limits to value in electronic commerce-related information technology investments. *Proceedings of the 33rd Annual Hawaii International Conference on Systems Sciences*, Wailea, Hawaii.

Choo, K.K.R. (2008) Organized crime groups in cyberspace: a typology. *Trends in Organized Crime*, *11*(3), 270-295.

Choo, K.K.R. & Smith, R.G. (2008). Criminal exploitation of online systems by organised crime groups. *Asian Journal of Criminology*, *3*(1), 37–59.

Chua, W.F. (1986). Radical developments in accounting thought. *The Accounting Review*, *61*(4), 601-632.

Chung, W., Chen, H., Chang, W., & Chou S. (2006). Fighting cybercrime: a review and the Taiwan experience. *Decision Support Systems*, *41*, 669-682.

Clayton, R. (2009). How much did shutting down McColo help? *Proceedings of the Sixth Conference on Email and Anti-Spam*, Mountain View, CA.

Cohen, F. (1987). Computer viruses: Theory and experiments. *Computers & Security*, *6*, 22–35.

Conti, G. (2000). Why computer scientists should attend hacker conferences. *Communications of the ACM*, *48*(3), 23-24.

Corey, V., Peterman, C., Shearin, S., Greenberg, M.S., & Van Bokkelen, J. (2002) Network forensics analysis. *IEEE Internet Computing*, *6*(6), 60-66.

D'Arcy, J. & Hovav, A. (2007). Deterring internal information systems abuse. *Communications of the ACM*, *50*(10), 113-117.

D'Arcy, J. & Hovav, A. (2009). Does one size fit all? Examining the differential effects of IS security countermeasures. *Journal of Business Ethics*, *89*, 59-71.

D'Arcy, J., Hovav, A., & Galletta, D. (2009). User awareness of security countermeasures and its impact on information systems misuse: a deterrence approach. *Information Systems Research*, *20*(1), 79-98.

Darke, P., Shanks, G., & Broadbent, M. (1998). Successfully completing case study research: Combining rigor, relevance, and pragmatism. *Information Systems Journal*, *8*, 273-289.

Dhillon, G. (1995). *Interpreting the Management of Information Systems Security*. (Doctoral dissertation). London School of Economics and Political Science.

Dhillon, G. & Backhouse, J. (1994). Responsibility analysis: a basis for understanding complex managerial situations. In *1994 International System Dynamics Conference*, Stirling, Scotland.

Dhillon, G. & Backhouse, J. (1996). Risks in the use of information technology within organizations. *International Journal of Information Management*, *16*(1), 65-74.

Dhillon, G. & Moores, S. (2001). Computer crimes: Theorizing about the enemy within. *Computers & Security*, *20*(8), 715-723.

Doss, G.W. (2012). *An Approach to Effectively Identify Insider Attacks within an Organization* (Doctoral dissertation). Retrieved from ProQuest Dissertations and Theses. (UMI No. AAT 3499545).

Dowland, P.S., Furnell, S.M., Illingworth, H.M., & Reynolds, P.L. (1999). Computer crime and abuse: a survey of public attitudes and awareness. *Computers & Security*, *18*(8), 715-726.

Dubé L. & Paré, G. (2003). Rigor in information system positivist case research: Current practices, trends, and recommendations. *MIS Quarterly*, *27*(4), 597-635.

Ebadi, Y.M. & Utterback, J.M. (1984). The effects of communication on technological innovation. *Management Science*, *30*(5), 572-585.

Eisenhardt, K.M. (1989). Building theories from case study research. *Academy of Management Review*, *14*(4), 532-550.

Everett, C. (2009). The lucrative world of cyber-espionage. *Computer Fraud & Security*, *7*, 5-7.

Feenan, D. (2002). Legal issues in acquiring information about illegal behaviour through criminological research. *British Journal of Criminology*, *42*, 762-781.

Ferrell, J. & Hamm, M.S. (Eds.). (1998). *Ethnography at the edge: crime, deviance, and field research*. Boston, MA: Northeastern University Press.

Fichman, R.G. (2001). The role of aggregation in the measurement of IT-related organizational innovation. *MIS Quarterly*, *25*(4), 427-455.

Fichman, R.G., & Kemerer, C.F. (1999). The illusory diffusion of innovation: An examination of assimilation gaps. *Information Systems Research*, *10*(3), 255-275.

Foltz, C.B. (2004) Cyberterrorism, computer crime, and reality. *Information Management & Computer Security*, *12*(2), 154-166.

Gable, G.G. (1994). Integrating case study and survey methods: an example in information systems. *European Journal of Information Systems*, *3*(2), 112-126.

Gal-Or, E. & Ghose, A. (2005). The economic incentives for sharing security information. *Information Systems Research*, *16*(2), 186-208.

Galbreth, M.R. & Shor, M. (2010). The impact of malicious agents on the enterprise software industry. *MIS Quarterly*, *34*(3), 595-612.

Garber, L. (1999). Melissa virus creates a new type of threat. *IEEE Computer*, *32*(6), 16–19.

Garg, A., Curtis, J., & Halper, H. (2003). Quantifying the financial impact of IT security breaches. *Information Management & Computer Security*, *11*(2), 74-83.

Genge, B. & Enachescu, C. (2015). Non-intrusive historical assessment of internet-facing services in the internet of things. *Proceedings of the 5th International Conference on Recent Achievements in Mechatronics, Automation, Computer Science and Robotics*, Kolkata, India.

Gibbert, M., Ruigrok, W., & Wicki, B. (2008). What passes as a rigorous case study? *Strategic Management Journal*, *29*, 1465-1474.

Goel, S. (2011). Cyberwarfare: Connecting the dots in cyber intelligence. *Communictions of the ACM*, *54*(8), 132-140.

Goel, S. & Shawky, H.A. (2009). Estimating the market impact of security breach announcements on firm values. *Information & Management*, *46*(7), 404-410.

Gordon, L.A. & Loeb, M.P. (2002). The economics of information security investment. *ACM Transactions on Information and Systems Security*, *5*(4), 438 – 457

Gordon, L.A., & Loeb, M.P. (2006). Budgeting process for information security expenditures. *Communications of the ACM*, *49*(1), 121-125.

Gordon, L.A., Loeb, M.P., & Lucyshyn, W. (2003). Sharing information on computer systems security: An economic analysis. *Journal of Accounting and Public Policy*, *22*(6), 461-485.

Gordon, L.A., Loeb, M. P., & Sohail, T. (2003). A framework for using insurance for cyber-risk management. *Communications of the ACM*, *46*(3), 81-85.

Gordon, L.A., Loeb, M.P., & Sohail, T. (2010). Market value of voluntary disclosures concerning information security. *MIS Quarterly*, *34*(3), 567-594.

Gordon, S. & Ma, Q. (2003). Convergence of virus writers and hackers: Fact or fantasy? *Symantec Security Response*. Retrieved March 5, 2013 from http://219.239.88.139/image20010518/238454.pdf.

Halfond, W.G., Viegas, J., & Orso, A. (2006). A classification of SQL-injection attacks and countermeasures. *Proceedings of the IEEE International Symposium on Secure Software Engineering*, 13-15, Washington DC, USA.

Hancock, B. (2002). Security crisis management - the basics. *Computers & Security, 21*(5), 397-401.

Hansen, J.V., Lowry, P.B., Meservy, R., & McDonald, D. (2007). Genetic programming for prevention of cyberterrorism through dynamic and evolving intrusion detection. *Decision Support Systems*, *43*(4), 1362-1374.

Highland, H. J. (1997). A history of computer viruses – Introduction. *Computers & Security*, *16*, 412–415.

Hill, K. (2013, September 4). The terrifying search engine that finds internet-connected cameras, traffic lights, medical devices, baby monitors and power plants. *Forbes*.

Retrieved October 20, 2015 from http://www.forbes.com/sites/kashmirhill/2013/09/04/shodan-terrifying-search-engine/.

Hoath, P. & Mulhall, T. (1998). Hacking: motivation and deterrence, part I. *Computer Fraud & Security*, *1998*(4), 16-19.

Hollinger, R.C. (1991). Hackers: Computer heroes or electronic highwaymen? *Computers & Society*, *21*(1), 6-17.

Hollinger, R.C. & Lanza-Kaduce, L. (1988). The process of criminalization: The case of computer crime laws. *Criminology*, *26*(1), 101-126.

Holt, T.J. (2007). Subcultural evolution? Examining the influence of on- and off-line experiences on deviant subcultures. *Deviant Behavior*, *28*(2), 171-198.

Holt, T.J. (2009). Examining the role of technology in the formation of deviant subcultures. *Social Science Computer Review*, *28*(4), 466-481.

Holt, T.J. (2013). Exploring the social organisation and structure of stolen data markets. *Global Crime*, *14*(2), 155-174.

Holt, T.J. & Copes, H. (2010). Transferring subcultural knowledge on-line: Practices and beliefs of persistent digital pirates. *Deviant Behavior*, *31*(7), 625-654.

Holt, T.J. & Lampke, E. (2010). Exploring stolen data markets online: products and market forces. *Criminal Justice Studies: A Critical Journal of Law, Justice, and Society*, *23*(1), 33-50.

Holt, T.J., Strumsky, D., Smirnova, O., & Kilger, M. (2012). Examining the social networks of malware writers and hackers. *International Journal of Cyber Criminology*, *6*(1), 891-903.

Hovav, A. & D'Arcy, J. (2004). The impact of virus attack announcements on the market value of firms. *Information Systems Security*, *13*(3), 32-40.

Hovav, A. & D’Arcy, J. (2003). The impact of denial-of-service announcements on the market value of firms. *Risk Management and Insurance Review*, *6*(2), 97-121.

Hu, Q., Zhang, C., & Xu, Z. (2012). Moral beliefs, self-control, and sports: Effective antidotes to the youth computer hacking epidemic. *Proceedings of the 45th Hawaii International Conference on System Sciences*, 3061-3070, Maui, Hawaii.

Huang, D.Y. (2013). Profit-driven abuses of virtual currencies. Retrieved November 2, 2015 from http://sysnet.ucsd.edu/~dhuang/get.php?f=huang-research-exam.pdf.

Huang, D.Y., Dharmdasani, H., Meiklejohn, S., Dave, V., Grier, C., McCoy, D., Savage, S., Weaver, N., Snoeren, A.C., & Levchenko., K. (2014). Botcoin: monetizing stolen cycles. *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, San Diego, CA.

Ianelli, N. & Hackworth, A. (2006). Botnets as a vehicle for online crime. *Proceedings of the International Conference on Forensic Computer Science*, Brasila, Brasil.

Jennings, D.F. & Lumpkin, J.R. (1992). Insights between environmental scanning activities and Porter's generic strategies: An empirical analysis. *Journal of Management*, *18*(4), 791-803.

Johnston, A.C. & Warkentin, M. (2010). Fear appeals and information security behaviors: an empirical study. *MIS Quarterly*, *34*(3), 549-566.

Jordan, T. & Taylor, P. (1998). A sociology of hackers. *The Sociological Review*, *46*, 757-780.

Kamluk, V. The botnet ecosystem. Kaspersky Labs. Retrieved October 17, 2015 from: http://latam.kaspersky.com/sites/default/files/knowledge-center/kl_botnet%20ecosystem.pdf.

Kannan, K., Rees, J., & Sridhar, S. (2007). Market reactions to information security breach announcements: An empirical analysis. *International Journal of Electronic Commerce*, *12*(1), 69-91.

Karahanna, E., Straub, D.W., & Chervany, N.L. (1999). Information technology adoption over time: A cross-sectional comparison of pre-adoption and post-adoption beliefs. *MIS Quarterly*, *23*(2), 183-213.

Karami, M. & McCoy, D. (2013). Understanding the emerging thread of DDoS-as-a-service. *Proceedings of the 6th USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET '13)*, Washington, DC.

Keil, M. (1995). Pulling the plug: Software project management and the problem of project escalation. *MIS Quarterly*, *19*(4), 421-447.

Klein, H.K. & Myers, M.D. (1999). A set of principles for conducting and evaluating interpretive field studies in information systems. *MIS Quarterly*, *23(1)*, 67-88.

Koller, C. (2010). *Diffusion of Innovation and Fraud in the Subprime Mortgage Market* (Doctoral dissertation). Retrieved from http://cech.uc.edu/content/dam/cech/programs/criminaljustice/docs/phd_dissertations/2011-2010/Koller%20Cynthia.pdf.

Kshetri, N. (2005). Pattern of global cyber war and crime: a conceptual framework. *Journal of Information Management*, *11*, 541-562.

Kshetri, N. (2006). The simple economics of cybercrimes. *IEEE Security & Privacy*, *4*(1), 33-39.

Lacoste, J. & Tremblay, P. (2003). Crime and innovation: A script analysis of patterns in check forgery. *Crime Prevention Studies*, *16*, 169-196.

Lee, A.S. (1989). A scientific methodology for MIS case studies. *MIS Quarterly*, *13*(1), 33-50.

Lee, J. & Lee, Y. (2002). A holistic model of computer abuse within organizations. *Information Management and Computer Security*, *10*(2), 57-63.

Leeson, P.T. & Coyne, C.J. (2005). The economics of computer hacking. *Journal of Law, Economics, & Policy*, *1*, 511-532.

Leydon, J. (2011, October 19). Report: Hacking forum is a cybercrime academy. *The Register*. Retrieved January 26, 2014 from http://www.theregister.co.uk/2011/10/19/hacking_forums_exposed/.

Li, M. (2004). An approach to reliably identifying signs of DDOS flood attacks based on LRD traffic pattern recognition. *Computers & Security*, *23*(7), 549-558.

Liebenau, J. & Backhouse, J. (1990). *Understanding information: an introduction*. London: Macmillan.

Lu, Y., Polgar, M., Luo, X., & Cao, Y. (2010). Social network analysis of a criminal hacker community. *Journal of Computer Information Systems*, *51*(2), 31-41.

Lua, R. & Yow, K.C. (2011). Mitigating DDoS attacks with transparent and intelligent fast-flux swarm network. *IEEE Network*, *25*(4), 28-33.

Lyda, R. & Hamrock, J. (2007). Using entropy analysis to find encrypted and packed malware. *IEEE Security & Privacy*, *2*, 40-45.

Mahmood, A., Siponen, M., Straub, D., Rao, H.R., & Raghu, T.S. (2010). Moving toward black hat research in information systems security: an editorial introduction to the special issue. *MIS Quarterly*, *34*(3), 431-433.

Mandiant. (2013). APT1: Exposing one of China's cyber espionage units. Mandiant Intelligence Center Report. Retrieved March 6, 2013 from http://intelreport.mandiant.com/Mandiant_APT1_Report.pdf.

Mason, M. (2010). Sample size and saturation in PhD studies using qualitative interviews. *Forum: Qualitative Social Research*, *11*(3), Article 8.

McIllwain, J.S. (1999). Organized crime: A social network approach. *Crime, Law, and Social Change*, *32*(4), 301-323.

Mercuri, R.T. (2003). Analyzing security costs. *Communications of the ACM*, *46*(6), 15-18.

Mirkovic, J. & Reiher, P. (2004). A taxonomy of DDoS attack and DDoS defense mechanisms. *ACM SIGCOMM Computer Communication Review*, *34*(2), 39-53.

Moore, G.C. & Benbasat, I. (1991). Development of an instrument to measure the perceptions of adopting an information technology innovation. *Information Systems Research*, *2*(3), 192-222.

Moore, T., Clayton, R., & Anderson, R. (2009). The economics of online crime. *Journal of Economic Perspectives*, *23*(3), 3-20.

Mukherjee, B., Heberlein, L.T., & Levitt, K.N. (1994). Network intrusion detection. *IEEE Network*, *8*(3), 26-41.

Mustonen-Ollila, E., & Lyytinen, K. (2003). Why organizations adopt information system process innovations: a longitudinal study using Diffusion of Innovation theory. *Information Systems Journal*, *13*(3), 275-297.

National Institute of Standards and Technology. (1995). An introduction to computer security: the NIST handbook. *Special Publication 800-12*, Retrieved February 16, 2013 from: http://csrc.nist.gov/publications/nistpubs/800-12/handbook.pdf

Nilakanta, S. & Scamell, R.W. (1990). The effect of information sources and communication channels on the diffusion of innovation in a data base development environment. *Management Science*, *36*(1), 24-40.

Oliner S.D. & Sichel, D.E. (2000). The resurgence of growth in the late 1990s: Is information technology the story? *Journal of Economic Perspectives*, *14*(4), 3-22.

Ollman, G. (2008). The evolution of commercial malware development kits and colour-by-numbers custom malware. *Computer Fraud & Security*, *2008*(9): 4-7.

Orlikowski, W.J. & Baroudi, J.J. (1991). Studying information technology in organizations: Research approaches and assumptions. *Information Systems Research*, *2*(1), 1-28.

Orman, H. (2003). The Morris Worm: A fifteen year perspective. *IEEE Security & Privacy*, *1*(5), 35–43.

Parekh, D.H., Dave, M.D., & Sridaran, R. (2014). Live Experiments depicting SQL Injection Attacks. *International Journal of Advanced Networking & Applications*, 91-93.

Parthasarathy, M. & Bhattacherjee, A. (1998). Understanding post-adoption behavior in the context of online services. *Information Systems Research*, *9*(4), 362-379.

Patel, A., Qassim, Q., & Wills, C. (2010). A survey of intrusion detection and prevention systems. *Information Management & Computer Security*, *18*(4), 277-290.

Peretti, K. (2009). Data breaches: what the underground world of carding reveals. *Santa Clara Computer & High Tech Law Journal*, *25*, 375-413.

Provos, N., Rajab, M. A., & Mavrommatis, P. (2009). Cybercrime 2.0: When the cloud turns dark. *Communications of the ACM*, *52*(4), 42–47.

Ransbotham, S. & Mitra, S. (2009). Choice and chance: a conceptual model of paths to information security compromise. *Information Systems Research*, *20*(1), 121-139.

Reich, B.H. & Benbasat, I. (1990). An empirical investigation of factors influencing the success of customer-oriented strategic systems. *Information Systems Research*, *1*(3), 325-347.

Richardson, R. (2011). 2010 / 2011 CSI Computer Crime and Security Survey. Retrieved January 27, 2013 from www.ncxgroup.com/wp-content/uploads/2012/02/CSIsurvey2010.pdf.

Rogers, E.M. (2003). *Diffusion of innovations* (5th ed.). New York, NY: Free Press.

Rossow, C. Amplification hell: Revisiting network protocols for DDoS abuse. *Proceedings of the Symposium on Network and Distributed System Security (NDSS*), San Diego, CA.

Rowley, J. (2002). Using case studies in research. *Management Research News*, *25*(1), 16-27.

Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, *14*(2), 131-164.

Sarker, S. & Lee, A.S. (2002). Using a positivist case research methodology to test three competing theories-in-use of business process redesign. *Journal of the Association for Information Systems*, *2*(7).

Sarker, S. & Lee, A.S. (2003). Using a case study to test the role of three key social enablers in ERP implementation. *Information & Management*, *40*, 813-829.

Schechter, S.E. (2005). Toward econometric models of the security risk from remote attacks. *IEEE Security & Privacy*, *3*(1), 40-44.

Schneier, B. & Kelsey, J. (1999). Secure audit logs to support computer forensics. *ACM Transactions on Information and System Security*, *2*(2), 159-176.

Schultz, E.E. (2002). A framework for understanding and predicting insider attacks. *Computers & Security*, *21*(6), 526-531.

Sekaran, U. & Bougie, R. (2010). *Research Methods for Business: A Skill Building Approach* (5th ed.). West Sussex, UK: John Wiley & Sons.

Sharma, N. & Rathore, V.S. (2012). Analysis of different vulnerabilities in auto teller machine transactions. *Journal of Global Research in Computer Science*, *3*(3), 38-40.

Shiaeles, S.N., & Papadaki, M. (2014). FHSD: an improved IP spoof detection method for web DDoS attacks. *The Computer Journal*, *58*(4), 892-903.

Snyder, P.J., Priem, R.L., & Levitas, E. (2009). The diffusion of illegal innovations among management elites. *Academy of Management Proceedings*, *2009*.

Stake, R.E. (1995). *The Art of Case Study Research*. Thousand Oaks, CA: Sage Publications, Inc.

Straub, D.W. (1990). Effective IS security: an empirical study. *Information Systems Research*, *1*(3), 255-276.

Straub, D. W. (1994). The effect of culture on IT diffusion: E-Mail and FAX in Japan and the US. *Information Systems Research*, *5*(1), 23-47.

Straub, D.W., Boudreau, M.C., & Gefan, D. (2004). Validation guidelines for IS positivist research. *The Communications of the Association for Information Systems*, *13*(1), 63.

Straub, D.W. & Nance, W.D. (1990). Discovering and disciplining computer abuse in organizations: a field study. *MIS Quarterly*, *14*(1), 45-60.

Straub, D.W. & Welke, R.J. (1998). Coping with systems risk: Security planning models for management decision making. *MIS Quarterly*, *22*(4), 441-469.

Stewart, K.J. & Gosain, S. (2006). The impact of ideology on effectiveness in open source software development teams. *MIS Quarterly*, *30*(2), 291-314.

Stone-Gross, B., Kruegel, C., Almeroth, K., Moser, A., & Kirda, E. (2009). FIRE: Finding rogue networks. *Proceedings of the 2009 Computer Security Applications Conference*, 231-240, Honolulu, HI.

Stuart, I., McCutcheon, D., Handfield, R., McLachlin, R., & Samson, D. (2002). Effective case research in operations management: a process perspective. *Journal of Operations Management*, *20*, 419-433.

Tejay, G.P.S. & Zadig, S.M. (2012). Investigating the effectiveness of IS security countermeasures towards cyber attacker deterrence. *Proceedings of the 45th Annual Hawaii International Conference on Systems Sciences*, Maui, Hawaii.

Tellis, W. (1997a). Introduction to case study. *The Qualitative Report*, *3*(2).

Tellis, W. (1997b). Application of a case study methodology. *The Qualitative Report*, *3*(3).

Theoharidou, M., Kokolakis, S., Karyda, M., & Kiountouzis E. (2005). The insider threat to information systems. *Computers & Security*, *24*, 472-484.

Thomas, R. & Martin, J. (2006). The underground economy: Priceless. *USENIX ;login*, *31*(6), 7-16.

Thong, J.Y.L. (1999). An integrated model of information systems adoption in small businesses. *Journal of Management Information Systems*, *15*(4), 187-214.

Turgeman-Goldschmid, O. (2005). Hacker's accounts: Hacking as social entertainment. *Social Science Computer Review*, *23*(8), 8-23.

Turgeman-Goldschmid, O. (2008). Meanings that hackers assign to their being a hacker. *International Journal of Cyber Criminology*, *2*(2), 382-396.

UN Office on Drugs and Crime. (2013). Comprehensive Study on Cybercrime (Draft). Retrieved March 1, 2013 from http://www.unodc.org/documents/organized-crime/UNODC_CCPCJ_EG.4_2013/CYBERCRIME_STUDY_210213.pdf.

US Department of Justice. (2010). Prosecuting Computer Crimes. Retrieved February 11, 2013 from www.justice.gov/criminal/cybercrime/docs/ccmanual.pdf.

Utterback, J.M. (1971). The process of technological innovation within the firm. *Academy of Management Journal*, *14*(1), 75-88.

Venter, H.S. & Eloff, J.H.P. (2003). A taxonomy for information security technologies. *Computers & Security*, *22*(4), 299-307.

Verizon (2010). Verizon Data Breach Investigations Report. Retrieved January 29, 2013 from http://www.verizonbusiness.com/resources/reports/rp_2010-data-breach-report_en_xg.pdf.

Wang, N., Liang, H., Zhong, W., Xue, Y., & Xiao, J. (2010). Resource structuring or capacity building? An empirical study of the business value of information technology. *Journal of Management Information Systems*, *29*(2), 325-367.

Weber, T. (2007, January 25). Criminals may overwhelm the Web. *BBC News*. Retrieved February 23, 2013 from http://news.bbc.co.uk/2/hi/business/6298641.stm.

Wei, C., Sprague, A., Warner, G., & Skjellum, A. (2010). Characterization of spam advertised web hosting strategy. *Proceedings of the Sixth Conference on Email and Anti-Spam*, Mountain View, CA.

Willemson, J. (2006). On the Gordon & Loeb model for information security investment. *Proceedings of the Fifth Workshop on the Economics of Information Security*, Cambridge, England.

Willison, R. (2000). Understanding and addressing criminal opportunity: the application of situational crime prevention to IS security. *Journal of Financial Crime*, *7*(3), 201-210.

Willison, R. (2006). Understanding the offender/environment dynamic for computer crimes. *Information Technology & People*, *19*(2), 170-186.

Willison, R. & Backhouse, J. (2006). Opportunities for computer crime: Considering systems risk from a criminological perspective. *European Journal of Information Systems*, *15*, 403-414.

Willison, R. & Warkentin, M. (2013). Beyond deterrence: an expanded view of employee computer abuse. *MIS Quarterly*, *37*(1), 1-20.

Xu, W., Nguyen, H., Grant, G., & Dai, X. (2008). Security breach: the case of TJX Companies, Inc. *Communications of the Association for Information Systems*, *23*(31), 575-591.

Yan, W., Zhang, Z., & Ansari, N. (2008). Revealing packed malware. *IEEE Security & Privacy*, *6*(5), 65-69.

Yin, R.K. (2009). *Case Study Research: Design and Methods* (4th ed.). London, UK: SAGE Press.

Young, R., Zhang, L., & Prybutok, V. (2007). Hacking into the minds of hackers. *Information Systems Management*, *24*(4), 281-287.

Zadig, S.M. & Tejay, G. (2010). Securing IS assets through hacker deterrence: A case study. *Proceedings of the 2010 IEEE eCrime Researchers Summit*, Dallas, TX.

Zadig, S.M. & Tejay, G. (2011). Emerging cybercrime trends: Legal, ethical, and practical issues. In A. Dudley, J. Braman, & G. Vincenti, (Eds.), *Investigating cyber law and cyber ethics: Issues, impacts, and practices*. Hershey, PA: IGI Global.

Zmud, R.W. (1983). The effectiveness of external information channels in facilitating innovation within software development groups. *MIS Quarterly*, *7*(2), 43-58.