

***Hacking Framework Extended:
The Role of Vulnerabilities***

Joseph H. Schuessler

University of North Texas
Joseph.Schuessler@unt.edu

Bashorat Ibragimova

Texas A&M University - Commerce
Bashorat_Ibragimova@tamu-commerce.edu

Abstract

Information Systems Security (ISS) has become a major concern in the United States following the rapid commercialization of the Internet, terrorism awareness after 9/11, and changes brought about by Sarbanes-Oxley and similar acts. This study explores the compromise of systems by extending the hacking framework (Bento and Bento, 2004). We apply user system vulnerabilities to the framework to investigate user compromise. The number of broadband connections and vulnerabilities were found to be significantly related to user compromise. Implications for practice are discussed as well as suggestions for future research.

Keywords: *Hacking, Information Warfare, Reconnaissance, Security, User Compromise, Vulnerabilities*

Introduction

Security is a major concern in Information Systems (IS) because it affects the integrity of data, the confidence people have in organizations, and the ability of employees to perform their job-related tasks in an efficient manner. But the nature of Information System Security (ISS) has changed dramatically over the years. ISS is considerably different from the time when employees worked at dumb terminals connected to a mainframe where the majority of threats were related to physical access, typically by the insiders of the company. ISS has evolved to the point where an outsider is considered a greater threat than an insider (Pfleeger and Pfleeger, 2002), mainly due to the ease of connectivity brought about by the Internet. This emerging view is in direct conflict with conventional wisdom and one possible explanation for this divergence in views on ISS is that "... information systems managers have migrated their organizations into the highly interconnected environment of modern technology but continue to view threats from a perspective of a pre-connectivity era" (Loch et al., 1992).

This disagreement on the origin of threats has led to an increased focus on prevention activities, often at the expense of deterrence, detection, and remedies (Straub and Welke, 1998) despite studies that have shown the effectiveness of each at increasing the effectiveness of ISS [10, 23]. Managers who are aware of a wide range of controls are more likely to use them and they are not predisposed to ignore theoretically driven security measures (Straub and Welke, 1998). But managers must not only be aware of various controls they have at their disposal in order to

mitigate risk, but also of the vulnerabilities inherent in their systems that contribute to that risk in the first place. If IS managers lack an understanding of the various vulnerabilities their IS infrastructure faces, they risk the financial well-being of their organizations.

In 2004, seventy one percent of attacks were initiated by outsiders to the organization (CERT/CC, 2004). It is possible that the focus of CERT/CC on vulnerabilities has led to a bias in their statistics, but the overall role of vulnerabilities cannot be overstated. Due to a large number of automated attack tools and the volume of Internet connections, the number of security incidents reported no longer provides quality information regarding the scope and impact of security incidents (CERT/CC, 2004). Many of the early works in ISS provide insight into the scope and role of ISS but much of it pre-dates the wide-spread commercialization of the Internet. Additionally, as information warfare migrates from the military arena to the commercial arena (Knapp and Boulton, 2006) demonstrating the dynamic nature of ISS, a continual reexamination of how security breaches occurs and spread is necessary.

Security breaches encompass more than just the compromise of sensitive company data and valuable hardware. There are real dollar implications. For example, of those surveyed in the 2007 CSI/FBI Computer Crime and Security Survey, security breaches average \$345,005 per respondent (Richardson, 2007) and overall financial losses of greater than \$666,000,000 were incurred in 2003 (CERT/CC, 2004). These figures probably underestimate the real volume of financial losses due to the unwillingness of respondents to disclose financial losses (Bagchi and Udo, 2003). Yet despite the costs associated with security breaches, some companies still question the need to invest in security technologies (Kros et al., 2004/2005).

Background and Research Framework

Given the significant volume of security breaches and the dollar impact, a better understanding of the nature of attacks is crucial. That is the purpose of this paper, to further refine our understanding of the salient issues as they relate to ISS. This study extends the framework developed by Bento and Bento (2004) by including an examination of vulnerabilities and the role they play in user compromise. The focal point of the Bento and Bento (2004) framework was the investigation of the underlying motivations of users launching denial of service (DoS) attacks. Exploratory in nature, that framework proposed that the number of hosts, the number of broadband connections, the volume of reconnaissance activities, and the volume of malicious code all lead to user compromise, root compromise, and ultimately to the launching of DoS attacks (Bento and Bento, 2004).

Bento and Bento (2004) did not include a vulnerability construct in their initial framework and thus left out a potentially significant construct that may impact several of the constructs of interest in their model. The significant contribution of the current research is incorporation of such a construct and a refocus of the model to examine the impact on user compromise. The following sections develop the hypotheses put forth in the current research. Several of these are taken directly from Bento and Bento (2004) in an effort to not only act in a confirmatory role, but to also add validity to the unique contributions of the current study.

Bento and Bento (2004) identified reconnaissance as a method hackers to use to “identify potential victims” and posit that the higher the number of broadband connections, the greater the amount of reconnaissance activities would occur (Hypothesis 1). This can be thought of from two perspectives. With an increase in the number of broadband connections, there are more avenues from which hackers can conduct reconnaissance activities. Conversely, with an increase in broadband activities, there are more potential “victims” about which reconnaissance activities can be conducted. In either case, the relationship should be positive.

Reconnaissance is used in a variety of ways: to identify potential victims, to assess vulnerabilities, and to identify the potentially applicable forms of malicious code that could later be used to compromise a user. Hypothesis 2 posits that a rise in reconnaissance will coincide with a rise in the use of malicious code. Though malicious code can be used for a variety of purposes, such as hijacking machines in order to display ads for vendors on target machines, it can also be used for monitoring services running on a target machine in order to identify and exploit its vulnerabilities. The fact that malicious code can be used to gather information about a target machine suggests that there is a positive relationship between malicious code and reconnaissance activities. Therefore, as we see a rise in malicious code, we should also see a rise in reconnaissance activities (Hypothesis 2).

The number of broadband connections was positively related to the number of user compromises reported according to Bento and Bento (2004). They based their hypothesis on existing literature which suggested that much of the growth of broadband connections had been attributed to less experienced computer users such as common household users and small businesses which could become ideal targets for potential hackers. Referred to as “resource poverty” (Thong et al., 1996), the idea is that such users often lack the resources necessary to effectively plan, develop, and implement a secure information infrastructure compared to those with more experience and more financial resources. This leads to hypothesis 3 in the current study which suggests that as the number of broadband connections increases, the more user compromises will occur.

As discussed above, reconnaissance activities refer to the identification and assessment of potential targets. Information such as the operating system, running services, and installed patches and so forth are important for potential attackers to know in order to figure out how to compromise a system. Therefore, the more reconnaissance activities that occur should lead to a higher likelihood of user compromise (Hypothesis 4).

The current study also suggests a positive relationship between malicious code and user compromise (Hypothesis 5). In addition to some of the various uses of malicious code discussed above, malicious code can also be used to capture keystrokes, user names and passwords. Keystroke loggers and packets sniffers can aid those intent on compromising a system by capturing this information and sending it to them after some predetermined time or event. After obtaining all the rights and privileges of a legitimate user, an attacker can then proceed to access, modify, and destroy data of the legitimate user or attempt to escalate their privileges in an attempt to gain administrative/root privileges.

Though Bento and Bento (2004) indirectly suggested that a relationship between the number of broadband connections and the volume of malicious code incidents exists in their correlational

analysis, the current study explicitly hypothesizes this relationship. Due to the self-replicating nature of the malicious code and its ability to spread via the Internet without the need for traditional reconnaissance activities or human intervention, there is likely to be a positive relationship between malicious code and the number of broadband connections (Hypothesis 6).

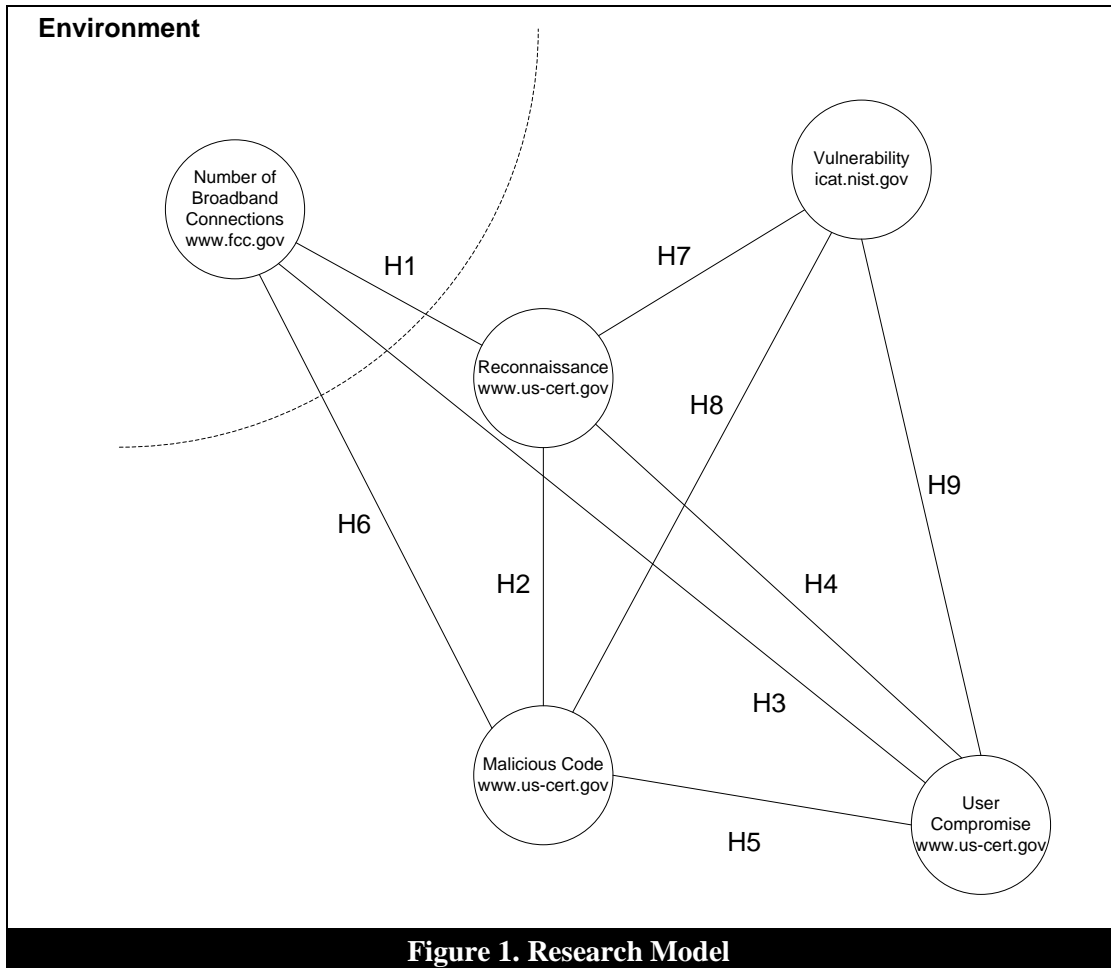
Also, while Bento and Bento (2004) specifically mention vulnerabilities, their research does not incorporate a vulnerability construct. This leads to hypotheses unique to the current study, namely, H7, H8, and H9. Greater levels of reconnaissance are likely to lead to a discovery of more vulnerabilities suggesting that there is a positive relationship. Conversely, the discovery and announcement of vulnerabilities is likely to be positively related to reconnaissance activities searching for targets with those vulnerabilities. Hence, H7 posits that there is a relationship between reconnaissance and vulnerabilities.

Malicious code is often created to take advantage of specific vulnerabilities in an operating system, specific applications, or some other inherent weakness in a target machine. Therefore, hypothesis 8 suggests that there is a positive relationship between malicious code and vulnerabilities. The higher the number of vulnerabilities in potential targets leads to the higher number of incidents involving the use of malicious code.

Finally, vulnerabilities provide a weakness in a target machine potentially leading to the compromise a user. So the higher the number of vulnerabilities increases the likelihood of a user being compromised. Hypothesis 9 states that higher number of system vulnerabilities will lead to an increase in the number of user compromises reported. Table 1 below provides a summary of all hypotheses, while Figure 1 illustrates the research model.

Table 1. Summary of Hypotheses

Hypothesis	
H1	There is a positive relationship between broadband connections and reconnaissance activities.
H2	There is a positive relationship between reconnaissance activities and malicious code incidents.
H3	There is a positive relationship between broadband connections and user compromise.
H4	There is a positive relationship between reconnaissance and user compromise.
H5	There is a positive relationship between malicious code and user compromise.
H6	There is a positive relationship between malicious code and broadband connections.
H7	There is a positive relationship between reconnaissance and system vulnerabilities.
H8	There is a positive relationship between system vulnerabilities and malicious code incidents.
H9	There is a positive relationship between system vulnerabilities and user compromise.



Research procedure

In order to examine the proposed relationships, archival data was gathered from several sources. These sources included the ICAT database maintained by the NIST, the Federal Communications Commission (FCC), and the United States Computer Emergency Readiness Team (CERT/CC) (2004). The use of archival data, though helpful, required consistent comparison periods. As such, the number of data points used for the analysis consisted of 53 months from February 2000 to June of 2004. Contributions from each of the data sources is discussed below.

For purposes of this study, the vulnerabilities construct was borrowed from the ICAT database maintained by the National Institute of Standards and Technology (NIST, 2005). Vulnerabilities are recorded in the database individually by date, vulnerability type, and by the product that the vulnerability relates to. The database classifies vulnerabilities into different vulnerability types including input validation errors, buffer overflow errors, boundary condition errors, access validation errors, exception condition handling errors, environmental errors, configuration errors, race condition errors, design errors, and other errors. For detailed definitions of each of these vulnerabilities, please refer to Appendix A. Because each vulnerability discovered was listed by its date of discovery, each type of vulnerability was summed for each month over the target time

frame as discussed above in order to get a monthly figure for each type of vulnerability. Each vulnerability type then served as a potential item for measuring the vulnerability construct.

The data for broadband connections were obtained from an FCC news release on high speed Internet access in the United States which is issued bi-annually. The term “high speed Internet access” is defined by the FCC as “...connections that deliver services at speeds exceeding 200 kilobits per second (kbps) in at least one direction...” (FCC, 2005). It should be noted that the FCC only gathers high-speed Internet access statistics over six-month periods. Therefore, for this construct, the six-month data was repeated for each of its respective monthly data points. Although this is a limitation of the study, this compromise it does capture the long-term trend of the number of broadband connections in the United States and was necessary in order to make monthly comparisons.

The reconnaissance construct data was gathered from US-CERT (2004). US-CERT is a sub-division of the National Cyber Security Division of the Department of Homeland Security. This sub-division coordinates defenses and responses with other organizations, such as the Department of Homeland Security and CERT/CC, in order to protect the information infrastructure of the United States. Defined as a combination of footprinting, scanning, and enumeration (McClure et al., 2001), reconnaissance activities refer to actions involving investigation of the target, identification of its weaknesses, and other key information, such as user account information. US-CERT gathers this information on incidents reported in the United States and compiles it into a monthly report listing the number of incidents that occur each month (2004).

US-CERT also gathers information on malicious code incidents. Malicious code refers to code used by hackers such as viruses, worms, and scripts to gather information, gain access to systems, or other actions that potentially violate a system. US-CERT reported these figures on a monthly basis from January 1999 to June 2004. Infections of malicious code can divert resources and disrupt business continuity (Logan and Logan, 2003) making inclusion of a malicious code construct necessary in the information system security model.

User compromise data were also gathered from US-CERT (2004). User compromise refers to gaining access to a legitimate user name and password, thus giving an attacker all the rights and privileges on the network that the legitimate user has. Like malicious code and reconnaissance data, user compromise data are compiled by US-CERT on a monthly basis.

With the exception of the vulnerabilities construct and the broadband construct, all other constructs were measured using a single item. A factor analysis was conducted on each multi-item construct constraining the number of factors for each to a single factor using SPSS version 13.0 and a varimax rotation. Each item for the broadband construct (DSL, Cable, Fiber, Satellite, and Other) loaded .966 or higher on its single factor and explained 96.698 percent of the variance with an R² of .639. Each type of vulnerability discussed above also loaded on a single factor and explained 49.914 percent of the variance with all items except Boundary Condition Errors and Other Errors loading at .566 or higher. As a result, these items were excluded from the subsequent PLS analysis discussed below. The R² for the vulnerabilities construct was .810.

The research model was assessed using SmartPLS version 2.0 (Ringle et al., 2005). PLS has been used in other security related research [10, 23]. It is a structural technique with unique characteristics that make it appropriate for analyzing many security related models. First, the ability of PLS to handle small sample sizes makes it a suitable choice due to characteristically small samples explored in security research (Kotulic and Clark, 2004). Second, PLS does not impose homogeneity or normality requirements on the data (Chin et al., 1996) as typically required with other structural techniques (Hair et al., 1998).

The model was analyzed to obtain the structural path coefficients and the construct variances. In order to obtain t-values for each path coefficient, 200 random samples of 100 were generated using a bootstrap procedure. Finally, the hypotheses were evaluated by assessing the sign and significance of the structural path coefficients using one-tailed t-tests. SmartPLS does not calculate any goodness-of-fit values. Rather, R2 values were evaluated to assess the ability of various proposed relationships to predict a significant degree of explanatory power in each construct. The results of the analyses are discussed in the next section.

Data Analysis

Table 2 below provides descriptive statistics and correlations for each construct. For purposes of the correlational analysis, the vulnerabilities construct was aggregated by summing each type of vulnerability for each month.

Table 2. Matrix of Intercorrelations among Constructs

	Mean	SD	User Compromise	Malicious Code	Recon.	Broadband Connections	Vulnerabilities Reported
User Compromise	20.98	74.331	1 .				
Malicious Code	34360.32	150205.13	-0.022 0.875	1 .			
Reconnaissance	1057785	7453037.8	0.002 0.988	.756* 0	1 .		
Broadband Connections	18000000	9763660.1	0.038 0.789	.836* 0	.850* 0	1 .	
Vulnerabilities Reported	125.6226	86.68	0.003 0.985	-0.078 0.581	0.026 0.853	-0.098 0.484	1 .

*. Correlation is significant at the 0.10 level (1-tailed).

To test the structural model, SmartPLS was used (Ringle et al., 2005). As discussed above, PLS does not provide fit indices or model assessments. The validity of the model is assessed by examining the significance of path coefficients and the amount of variance of the respective constructs that is explained. The significance level for evaluating each hypothesis was set at ten percent. This level is appropriate given the exploratory nature of the study.

The first hypothesis stated that the number of broadband connections would be positively related to the amount of reconnaissance reported. This relationship was significant and positive but only about 5% of the variance in reconnaissance was explained. Though the result is similar to that found by Bento and Bento (2004), it suggests that there are other factors in play that perhaps can better predict the amount of reconnaissance activities that take place. Never the less, this finding suggests that the higher the number of broadband connections, the more that reconnaissance activities occurs. As discussed earlier, this can be interpreted in two ways. First, the higher the number of broadband connections could simply represent an increase in potential “victims” of reconnaissance activities. Second, the increase in broadband connections could provide more avenues for people capable of committing reconnaissance activities. As broadband connections become increasingly affordable, an individual’s exposure is likely to be higher and the proliferation of “script kiddies” is likely to continue.

The second hypothesis stated a positive relationship between reconnaissance activities and malicious code. This hypothesis was also supported. As organizations see an increase in reconnaissance activities, they should make an effort to protect their systems by patching their operating systems, updating their anti-virus solutions, and monitoring their firewalls and intrusion detection systems in order to lower the risk of the impending spread of malicious code. Though malicious code was not found to lead to user compromise, it can impact systems in other ways such as slowing system performance and destroying or modifying data.

The third relationship, posited in the third hypothesis, was between the number of broadband connections and the number of user compromises. The results of the PLS analysis indicate a significant positive relationship between the number of broadband connections and user compromises. The higher number of broadband connections leads to more user compromises. This finding is consistent with the findings of Bento and Bento (2004). With the number of broadband connections predicted to continue to rise (IDC, 2006), we should continue to see increases in the number of user compromises.

The fourth hypothesis stated that the number of reconnaissance activities would be positively related to the number of user compromises reported. The results of the PLS analysis failed to support this hypothesis. This is an interesting result because intuitively one would expect hackers to conduct reconnaissance activities in order to identify and assess targets and ultimately compromise those systems. It appears that there is no relationship between the volume of reconnaissance and user compromises. One possible explanation of this counterintuitive finding might be the availability and use of automated attack tools that enable potential hackers to attack a large group of users without the need for reconnaissance information.

The fifth hypothesis stated that there was a positive relationship between malicious code and user compromise. This hypothesis was not supported at the .10 level of significance. So, even

though the amount of malicious code is expected to increase, there may not necessarily be an increase in user compromises due to the spread of malicious code. The result is similar to findings by Bento and Bento (2004) where malicious code was not found to be significantly related to user compromise. This may be due to distinctions in the types of malicious code where the goals of each type of malicious code are completely different. While some distinctions are rather obvious such as worms and bots, these distinctions often focus on the method of distribution rather than the payload of the code. Such a classification scheme which focused more on the intent of malicious code could potentially shed some light on this particular hypothesis.

The sixth hypothesis stated that the number of broadband connections would be positively related to the number of malicious code incidents reported. The PLS output supports this proposed relationship. Therefore, as the number of broadband connections continues to increase, an increase in the spread of malicious code will follow. This relationship was not explored in the Bento and Bento (2004) article but given the high correlation found in that article, the model in the current study was modified to include this relationship. Worldwide broadband growth is expected to nearly double over the next five years (IDC, 2006). As a result of the anticipated growth in broadband connections, growth in malicious code has to be anticipated as well. Malicious code is not only a potential security problem, but is also a performance problem because malicious code consumes precious computing resources required for performing legitimate tasks.

The seventh hypothesis proposed that an increase in the volume of reconnaissance would be positively related to the number of vulnerabilities reported. The PLS analysis showed no significant relationship between these two constructs though the p-value was close at .103. This is interesting given the emphasis in industry and academia regarding the discovery and reporting of vulnerabilities. Even more interesting is that the directionality of the hypothesized relationship is opposite of the finding suggesting that there is an inverse relationship between reconnaissance activities and the number of vulnerabilities reported each month. A possible contributing factor could be the temporal relationship between when a vulnerability is discovered through reconnaissance activities and when it is reported. Events such as zero-day exploits could mask some of the effects of reconnaissance on the reporting of vulnerabilities (Porter, 2006). In his article, Porter (2006) discusses private efforts to purchase discovered vulnerabilities in order to keep them contained until an effective solution can be developed. Conversely, often times, exploits are developed for vulnerabilities long after patches have been issued. Users simply fail to update their systems and as a result, are vulnerable long after a patch has been issued. Reconnaissance activities could then be used to identify un-patched systems and target them specifically.

The eighth hypothesis stated that there would be a positive relationship between malicious code and vulnerabilities. Though the p-value was close at .117, a statistically significant relationship was not found. Similarly to the seventh hypothesis though, the relationship was negative, implying that an increase in the number of vulnerabilities may lead to a decrease in malicious code. Similar explanation could be possible. Zero day exploits or the failure to patch systems well after patches are released could be creating a temporal issue with respect to the statistics gathered for each of these constructs. Though zero day exploits would be unavoidable from an

end user's perspective, active patching of their systems could potentially eliminate the threat of suffering from malicious code that comes out after vulnerabilities have been discovered and for which patches have been created.

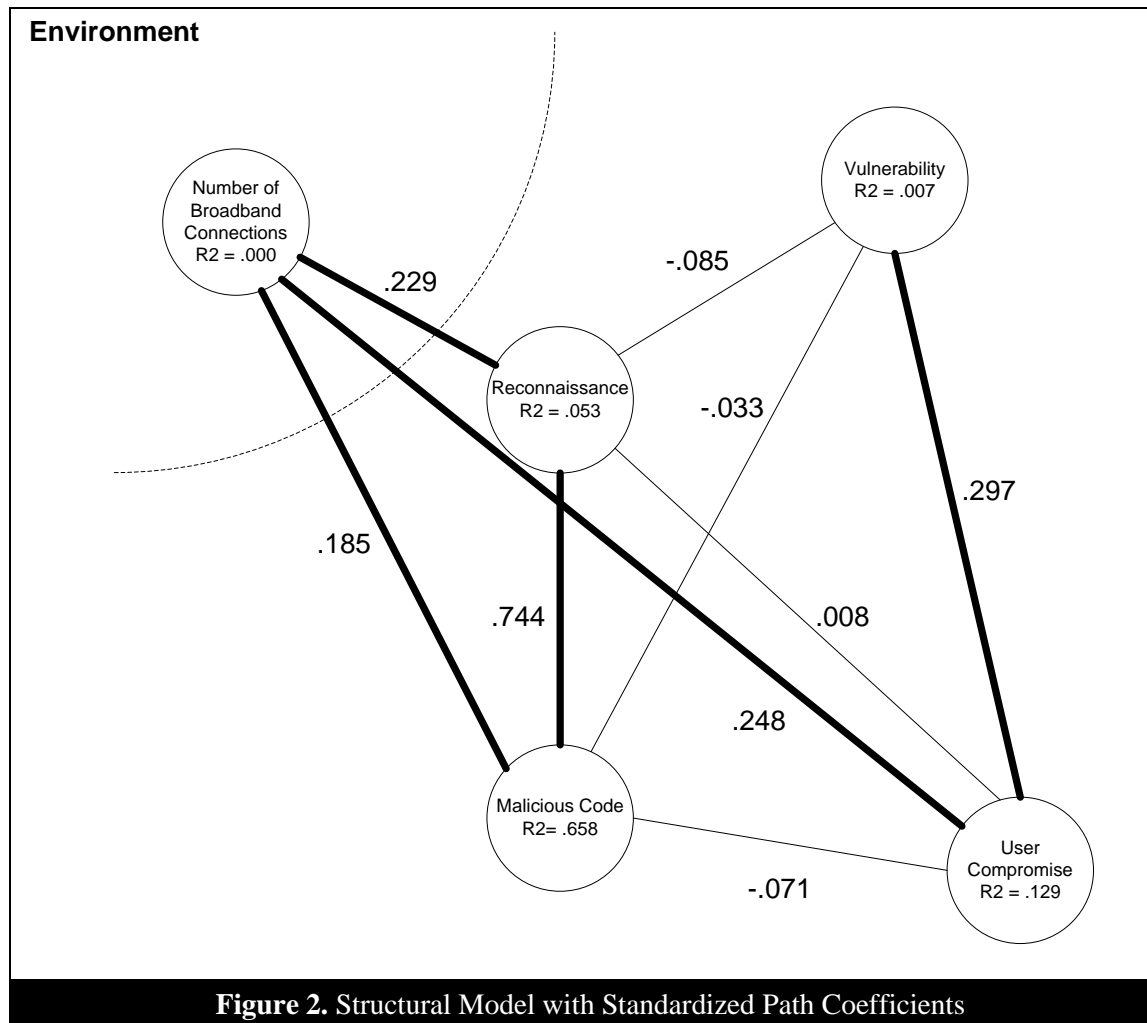
The final hypothesis stated that vulnerabilities would be positively related to user compromise. This hypothesis was supported suggesting that the higher the number of vulnerabilities, the higher the number of user compromises. This provides support for the importance of frequent system updates and antivirus solution installations in order to limit the potential compromise of user systems. Failure to eliminate (or at least reduce) user exposure to vulnerability can lead to user compromise, potential data loss, identity theft, and other privacy related issues. In light of the discussion above regarding zero day exploits, this finding would also suggest that hardware/software manufacturers should make a greater effort at eliminating vulnerabilities before releasing their products to begin with.

Results of the hypotheses are summarized in Table 2. The path model with standardized coefficients is presented in Figure 2. The model is assessed by examining the amount of variance explained by the various constructs. Given that user compromise was the focal point of the current research, the inclusion of the vulnerabilities construct improved the research model significantly by explaining 12.9 percent of the variance as compared to Bento and Bento (2004) where only 3.0 percent of the variance was explained. The contribution of this paper to research is illustrating the role that vulnerabilities play with user compromise and the other constructs within the model. The significance of the relationship between vulnerabilities and user compromise helps to further refine our understanding of each in the greater context of ISS as a whole.

Table 3. Results of the Hypotheses Testing

Hypotheses	Independent Variable	Dependent Variable	Standardized path Coefficient	Significance Level	Support $\alpha = .10$
H1	Broadband Connections	Reconnaissance	.229	.027	√
H2	Reconnaissance	Malicious Code	.744	.008	√
H3	Broadband Connections	User Compromise	.248	.010	√
H4	Reconnaissance	User Compromise	-.008	.491	X
H5	Malicious Code	User Compromise	-.071	.425	X
H6	Broadband Connections	Malicious Code	.185	.022	√
H7	Reconnaissance	Vulnerabilities	-.085	.103	X
H8	Vulnerabilities	Malicious Code	-.033	.117	X
H9	Vulnerabilities	User Compromise	.297	.050	√

Note: **Bold** entries indicate hypotheses unique to the current study.



Discussion and Conclusions

Some of the results, as expected, were similar to those found in Bento and Bento (2004), such as the relationship between reconnaissance and malicious code incidents. The inclusion of the relationship between malicious code and broadband connections, as well as the inclusion of the vulnerability construct were the major contributions of this paper. Using a similar dataset, the Bento and Bento (2004) research resulted in just over 3 percent of the variance being explained in user compromise. The current study, with the inclusion of the vulnerabilities construct accounted for 12.9 percent of the variance. Though significantly better, this finding suggests that there might be other relevant constructs that could improve the research model.

The hypotheses using the vulnerabilities construct provide valuable insights for practitioners and academicians alike. Boundary Condition Errors and Other Errors represented an insignificant amount of variance in the vulnerabilities construct. This suggests that some individual types of vulnerabilities have varying degrees of effects on the related constructs of interest. Therefore, an examination of the role that each type of vulnerability has on user compromise could shed additional light on the subject.

In terms of future research, the current study illustrates the need to extend the model in several directions. Further insight could be gained by trying to determine the salient issues as they relate to reconnaissance. Though the analysis suggested a significant relationship involving reconnaissance, the lack of explanatory power begs the question “What factors drive reconnaissance activities?” Additionally, the suggestion of the existence of a negative relationship between vulnerabilities and malicious code and vulnerabilities and reconnaissance raises questions as to how exactly hackers are using each technique to exploit users’ systems. A process model could help to illustrate the temporal relationship between each of these construct and help illustrate order and dependencies that might exist.

With the widespread growth of Internet and the need to connect remote systems to secure systems to facilitate e-commerce, security will continue to be of utmost concern to all involved. Hence, there is a need to understand how pertinent constructs interact with user compromise. The findings of this research suggest that greater care needs to be exercised in system configuration in order to reduce vulnerabilities that might lead to user compromise. The research model needs further refinement because some constructs were not as significant as expected. The effect of broadband type (cable, DSL, fiber, and so on) can be examined to determine if one type of broadband service is more likely to lead to user compromise than another. The reconnaissance construct could also be deconstructed in an attempt to separate locations of reconnaissance activities from those of their target systems. Additionally, a source of actual monthly data for the broadband statistics may need to be included in future research. Common method variance – obtaining all of the data from FCC – may account for temporal masking of some of the relationships proposed. Using other sources of relevant information may provide more interpretable results. One potential source might be Information Sharing and Analysis Centers (ISACs) that were established by the Presidential Decision Directive 63 that launched organizations designated to share information regarding vulnerabilities and security breaches (Gal-Or and Ghose, 2005). Device failures may also become a potential source for security vulnerabilities (Rae and Fidge, 2005). Other constructs could be added to determine the security effects that operating systems and device types have on user compromise. Lastly, the nomological aspects of the constructs should be further investigated in order to better determine what other constructs might be relevant and how they relate to those in the current study. The list of potential constructs might include legal construct addressing potential legal ramifications for hackers or the motivations driving hackers and/or the roles they are playing. Previous research has shown that hackers may pretend to be playing roles of computer security expert, law enforcement agent, intelligence agent, vigilante, organized crime member, hacktivist, or script kiddy (Falk, 2005). As we continue to grow in our understanding of information systems security, it has become clear that a constant reevaluation of ISS concepts is necessary in order to develop and maintain secure systems.

Appendix A. Definitions of vulnerabilities

Input validation error: Vulnerability is characterized as an “Input validation error” if the input being received by a system is not properly checked such that vulnerability is present that can be exploited by a certain input sequence. This vulnerability type and its subcategories only apply to input that is malicious or otherwise malformed. The “Input validation error” label may appear by itself or in two other variations: “Input validation error (Boundary overflow)” and “Input validation error (Buffer overflow)”. These two categories are discussed below.

Input validation error (Boundary overflow): A vulnerability is characterized as a “Boundary overflow” when the input being received by a system, be it human or machine generated, causes the system to exceed an assumed boundary thereby causing a vulnerability. For example, the system may run out of memory, disk space, or network bandwidth. Another example is that a variable might reach its maximum value and roll over to its minimum value. Yet another example is that the variables in an equation might be set such that a division by zero error occurs. Boundary overflow errors are a subset of the class of input validation errors. While it could be argued that buffer overflow (discussed next) is a type of boundary overflow error, we put buffer overflow in a distinct category given its importance.

Input validation error (Buffer overflow): Vulnerability is characterized as a “buffer overflow” if the vulnerability is caused by input being received by a system that is longer than the expected input length. If the system does not check for this condition then the input buffer fills up and overflows the memory allocated for the input. By cleverly constructing this extra input, an attacker can cause the system to crash or even to execute instructions on behalf of the attacker.

Access validation error: Vulnerability is characterized as “Access validation error” if a system is vulnerable because the access control mechanism is faulty. The problem lies not with the user controllable configuration of the access control mechanism but with the mechanism itself.

Exceptional condition handling error: Vulnerability is characterized as an “Exceptional condition handling error” if a system somehow becomes vulnerable due to an exceptional condition that has arisen. The handling (or mishandling) of the exception by the system enables a vulnerability.

Environmental error: Vulnerability is characterized as an “Environmental error” if the environment in which a system is installed somehow causes the system to be vulnerable. This may be due, for example, to an unexpected interaction between an application and the operating

system or between two applications on the same host. Such a vulnerable system may be perfectly configured and provably secure in the developers test environment, but the installation environment somehow violates the developer's security assumptions.

Configuration error: Vulnerability is characterized as a “Configuration error” if user controllable settings in a system are set such that the system is vulnerable. This vulnerability is not due to how the system was designed but on how the end user configures the system. We consider it a configuration error when a systems ships from a developer with a weak configuration.

Race condition: Vulnerability is characterized as a “Race condition” if a the non-atomicity of a security check causes the existence of a vulnerability. For example, a system checks to see if an operation is allowed by the security model and then performs the operation. However, between the time the security check is performed and when the operation is performed, the environment changes such that the operation is no longer allowed by the security model. Attackers can take advantage of this small window of opportunity and convince systems to perform illegal operations like writing to the password file.

Design error: Vulnerability is characterized as a “Design error” if there exists no errors in the implementation or configuration of a system, but the initial design causes a vulnerability to exist.

Other: Since the above vulnerability characteristics are not a true classification scheme, it is possible that vulnerability will not fall in any of them. Any such vulnerability is characterized as vulnerability type “Other” (NIST, 2005).

References

K. Bagchi, G. Udo, An analysis of the growth of computer and Internet security breaches, *Communications of the Association for Information Systems* (12), 2003, pp. 684-700.

A. Bento, R. Bento, Empirical test of hacking framework: an exploratory study, *Proceedings of the Tenth Americas Conference on Information Systems*, New York, New York, 2004, pp. 4526-4535.

CERT/CC, 2004 E-Crime watch survey shows significant increase in electronic crimes, <http://www.cert.org/archive/pdf/2004eCrimeWatchSummary.pdf> (Accessed 3-30-2005)

W. W. Chin, B. L. Marcolin, P. R. Newsted, A partial least squares latent variable modeling approach for measuring interaction effects: results from a Monte Carlo simulation study and voice mail emotion/adoption study, in *Proceedings of the Seventeenth International Conference on Information Systems*, J. I. DeGross, S. Jarvenpaa, and A. Srinivasan (eds.), 1996, pp. 21-41.

C. Falk, Ethics and hacking: the general and the specific, *Journal of Information Assurance*, 1(1), 2005, pp. 1-10.

FCC, Federal communications commission releases data on high-speed Internet access services, Federal Communications Commission, http://www.fcc.gov/Bureaus/Common_Carrier/Reports/FCC-State_Link/IAD/hspd1204.pdf (Accessed 3-25-2005).

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

J. F. Hair, R. E. Anderson, R. L. Tatham, W. C. Black, *Multivariate Data Analysis*, 5th edition, Pearson Education, Inc., Delhi, India, 1998.

IDC (2006) IDC forecasts worldwide broadband subscriptions to nearly double in five years, <http://www.idc.com/getdoc.jsp?containerId=prUS20208806> (Accessed 2-1-2007)

K. J. Knapp, W. R. Boulton, Cyber-warfare threatens corporations: expansion into commercial environments, *Information Systems Management*, 23(2), 2006, pp. 76-87.

A. G. Kotulic, J. G. Clark, Why there aren't more information security research studies. *Information & Management*, 41(5), 2004, pp. 597-607.

J. R. Kros, C. B. Folz, C. L. Metcalf, Assessing and quantifying the loss of network intrusion, *Journal of Computer Information Systems*, 45(2), 2004/2005, pp. 36-43.

K. D. Loch, H. H. Carr, M. E. Warkentin, Threats to information systems: today's reality, yesterday's understanding, *Management Information Systems Quarterly*, 16(2), 1992, pp. 173-186.

P. Y. Logan, S. W. Logan, Bitten by a bug: a case study in malware infection, *Journal of Information Systems Education*, 14(3), 2003, pp. 301-305.

S. McClure, J. Scambray, F. Kurtz, *Hacking Exposed*, McGraw-Hill, New York, 2001.

NIST (National Institute for Standards and Technology), C. S. D. The ICAT project, <http://icat.nist.gov> (3-23-2005), 2005.

C. P. Pfleeger, S. L. Pfleeger, *Security in Computing*. Prentice-Hall, Upper Saddle River, 2002.

B. Porter, Approaching zero: a study in zero-day exploits origins, cases, and trends, *Journal of Information Assurance*, 2(2), 2006, pp. 1-28.

A. Rae, C. Fidge, Information flow analysis for fail-safe devices, *The*

Computer Journal, 48(1), 2005, pp. 17-26.

R. Richardson, 2007 CSI/FBI computer crime and security survey, Computer Security Institute.

C. M. Ringle, S. Wende, A. Will, SmartPLS, Version 2.0 (beta). Hamburg, Germany, 2005.

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

J. Y. L. Thong, C.-S. Yap, K. S. Raman, Top management support, external expertise and information systems implementation in small businesses, Information Systems Research, 7(2), 1996, pp. 248-267.

US-CERT, Statistics on Federal incident reports, <http://www.uscert.gov/federal/statistics/>, 2004 (Accessed 3-17-2005)