Journal of International Technology and Information Management

Volume 26 | Issue 4 Article 3

12-1-2017

An Integrative Model of Managing Software Security during Information Systems Development

Vijay Raghavan
Northern Kentucky University, raghavan@nku.edu

Xiaoni Zhang
Northern Kentucky University, zhangx@nku.edu

Recommended Citation

Raghavan, Vijay and Zhang, Xiaoni (2017) "An Integrative Model of Managing Software Security during Information Systems Development," *Journal of International Technology and Information Management*: Vol. 26: Iss. 4, Article 3.
DOI: https://doi.org/10.58729/1941-6679.1335
Available at: https://scholarworks.lib.csusb.edu/jitim/vol26/iss4/3

An Integrative Model of Managing Software Security During Information Systems Development

Vijay Raghavan
(*Northern Kentucky University*)
raghavan@nku.edu

Xiaoni Zhang
(*Northern Kentucky University*)
zhangx@nku.edu

ABSTRACT

This study investigates the critical relationship between organizational system development policies, procedures and processes and the resulting security quality of the systems developed. We draw from a general software quality model to provide a theoretical foundation for testing this relationship. We used paper-based survey as well as online surveys to collect data from software developers and project managers. Our results revealed a significant relationship between management support and security policies and development process control. We also found significant relationships between development-process control and security quality, attitude and security quality, and the interaction between value congruence and commitment to provide security skills development. Counter-intuitively, we did not find a significant relationship between either security policy and security quality or the interaction between security policy and its legitimacy as perceived by systems development personnel. The managerial implications of the study include the need to foster a climate of security skills development through training for system development personnel and also simultaneously find strategies to more closely align their values to the security goals of the organization. Additionally, providing management support to formulate guidelines for development process control can improve the security quality of the systems developed.

KEYWORDS: Systems Development, Development Process Control, Security Quality, Management Support, Attitude

INTRODUCTION

There are many facets of managing security in information systems (IS). Although the question of how to build secure code from an architectural standpoint has been studied before (Ryoo, Kazman & Anand, 2015; Sood, 2012), the relationship between the organization context of systems development policies and procedures and the security of the systems developed has received only limited attention. In general, computer security has recently received greater attention (Schumacher et al., 2013). Amidst this increased attention to the general security-related issues, the emphasis on software development security has often taken the back seat to other pressing considerations such as delivering a product within budget and within the promised time frame.

Security as commonly understood refers to the ability of a system to protect information and system resources with respect to confidentiality and integrity. The vulnerabilities in a software product are first discovered by hackers who actively look for them rather than legitimate users. Many industrial groups and companies have developed security tools to enforce security features during application development process. The awareness of developers to security considerations along with an organization's focus on delivering a secure product often determines security quality of applications. There is evidence that bad coding practices create severe security problems (Denning, 2015; Lipner 2015). Being proactive during application development lifecycle is often considered the best approach to address security (Howard & Lipner, 2006).

Many agree that security must be designed and built into applications development life cycle (Wang & Wang, 2003). But, unfortunately, many of the security guidelines of regulations and standards can be bypassed while developing software. Deficient software development techniques and processes, a lack of security-focused quality assurance, and scarce security training for software developers, software architects, and project managers are often the unwitting culprits (Workman, Bommer & Straub, 2008). Jones and Rastogi (2004) clearly delineate the role of secure coding practices as a component of secure software development lifecycle. Their proposed life cycle explicitly includes many other elements of secure practices including security team review, data mutation and least privilege tests, penetration testing, external and internal certifications and security training and monitoring. It has been pointed out that one of the guiding principles of security management is to ensure that people understand their responsibility as well as their individual roles in establishing a secure information system (Dhillon & Backhouse, 2000). Developers are an important link in software security. And yet, it is very likely that some developers may not fully

comprehend as to how the software they're building or maintaining could be exploited in future (Van Wyk & Steven, 2006). In essence, security problems are people problems. Organizations often prefer to allocate resources to other areas of software development as the marginal returns on security investments are not easily quantifiable. A major reason for this mistaken emphasis might be that a secure application does not display its virtues, as for example an elegant user-interface might.

Improving the capability of the system development process has emerged as an important strategy for addressing recurring problems in software development, such as poor quality, high development costs, and long delivery lead times (Ravichandran & Rai, 2003). The proliferation of web applications has introduced many security holes. Web application code is the major reason for a website becoming vulnerable (Scott and Sharp 2002). Despite increased security interest in industries, research on application development security is sparse.

Therefore, developing an enhanced understanding of secured code or applications provides timely information to further our knowledge of software security. Security quality refers to addressing security threats and risks through software design factors (Wang & Wang, 2003). Our goal in this study to further the understanding of application-development security by 1) making an original effort in examining security in application development context, 2) proposing an integrative model that combines organizational level factors with individual factors for improving security quality and 3) testing the proposed model.

We draw from Ravichandran and Rai's (2000) model of general software quality and apply it to the context of security quality. They view the software quality management as an organizational system design endeavor and conceptualize its meaning consisting of both product quality and process efficiency. Our application of this general software quality mode is reasonable since security quality is only a subset of software quality that this study explores in greater depth. Our goal is to study how to manage the system development process and hence our emphasis has been on the process management constructs.

Our study also borrows from the leadership theory to apply security practices in application development. This theory suggests that by making followers more aware of the importance and value of the security, the participants in the system development process will be more responsive and sensitive to the potential vulnerability and risks associated with application development. As the importance of employees' following their organizations' information security rules and regulations increases, our study sheds light on the role of information

security awareness (ISA) and compliance-related beliefs in an organization's efforts to encourage compliance.

THEORETICAL FOUNDATIONS AND HYPOTHESES

There is a large body of literature on application development in general (Ravichandran & Rai, 2000 ; Howard & Lipner, 2006; Čeke and Milašinović, 2015; Lu et al., 2015; Barragáns-Martínez et al, 2015); however, only a few studies focus on how software development procedures and policies can affect application security (Bergvall-Kåreborn & Howcroft, 2014). Previous research results show that the deployment of methodologies by IS (Information Systems) developers is primarily associated with a hierarchical culture that is oriented toward security, order, and routinization (Iivari & Huisman, 2007). In this section, we address the theoretical foundations and results from key previous studies for the constructs used in our study.

RESEARCH MODEL

Our research model is based on Ravichandran and Rai's (2000) general model of software quality performance. Their model identifies critical organizational levers that IS managers can use in their efforts to improve software quality performance as well as employee specific factors that contribute to software development quality. Our model is presented in Figure 1 below and incorporates elements of leadership, structure, process, and outcome constructs specific to the software security implementation. The way our model maps to Ravichandran & Rai (2000) general software quality model is shown in the following diagram. In addition, we have included in our model legitimacy and value congruence two personal factors that are deemed to influence rule compliance behavior (Tyler & Blader, 2005).

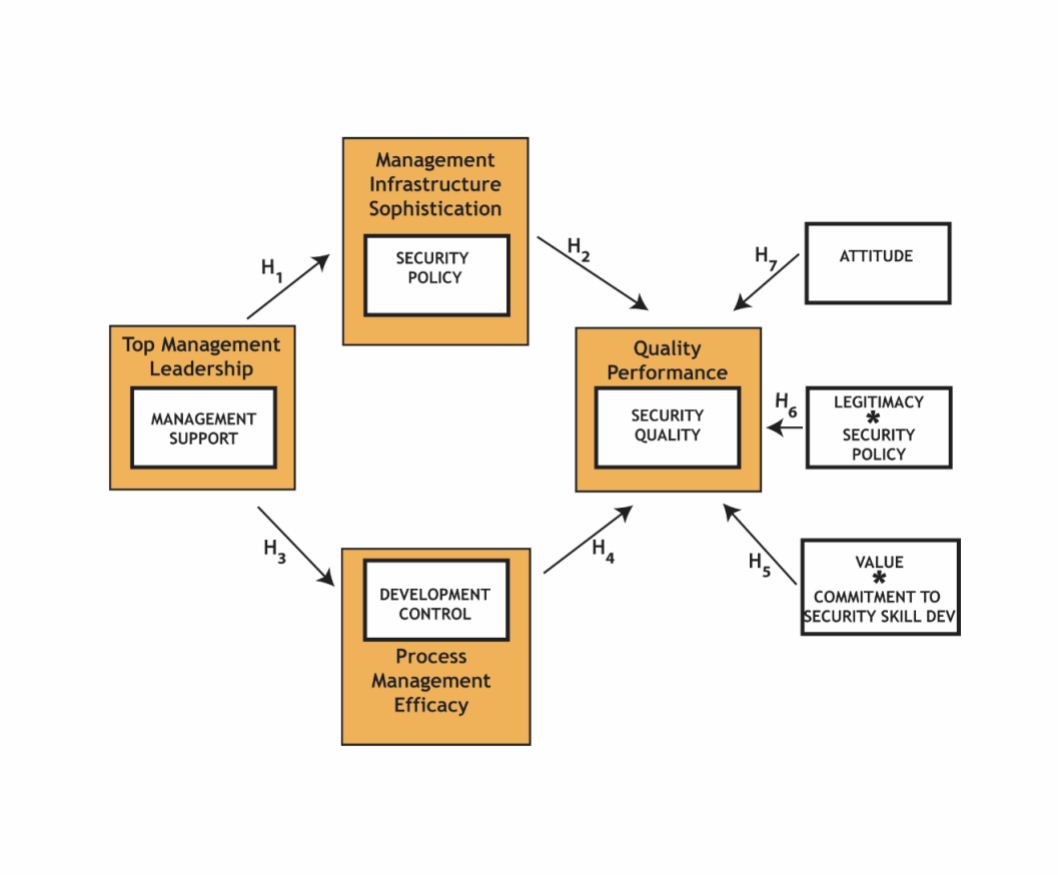

Figure 1. A Model of Implementing Software Security during Systems Development

The following section explains the model and a discussion of the various constructs and their rationale for inclusion in the model.

Security quality

Secure applications are generally accepted to be those that capture user input accurately, perform business functions correctly, and resist application breaches. In common practice, we see that at times applications do not enforce data validation; some do not function as expected, and others cannot secure data.

Insufficient security features in application development may cause huge losses for companies. Software developers occasionally form illusive trust assumptions and because of such assumptions, software development efforts often do not address potential security consequences adequately (Viega & Mcgraw, 2001). Common faults in coding such as lack of validation for user inputs, untested and inappropriate file calls leave security holes, and consequently, application vulnerabilities occur (King, 2004). Some security flaws are built into systems from the earliest stages of development due to insufficient awareness of security problems (Landwehr et al, 1994). Prior research on security has provided a useful starting point for our current study. Nabi (2005) examines e-commerce security and suggests strategies for secure business application logic: good design and engineering, secured configuration, defensive programming and secured wrappers for server-side software. Adams & Blandford (2005) advocate the understanding of communities of practice to enforce security and privacy issues within organizations. A socio-technical approach to security has been advocated to achieve three objectives: balancing the need to secure information assets against the need to enable the business, maintaining compliance, and ensuring cultural fit (Kayworth & Whitten, 2010). Among practitioners, Wang and Wang (2003) discuss security and quality issues related to software development and identify security risks and discuss the impact of security risks on quality factors. There is a general agreement that security procedures and practices must be instilled in all phases of application development. Training of developers and the support from the management are also important factors in developing secure systems (Popa, 2009).

Management support

Strategic leadership theory suggests that the CEOs' decisions and behavior are likely to explain organizational outcomes (Boeker, 1992; Allen & Panian, 1982). Prior works emphasize that top management leadership is an important and critical factor in general quality improvement (Deming 1986; Schonberger 1984). Top management's commitment to security can be shown in different forms: vision, mission, and value. By advocating the importance of security, organizational learning and culture are becoming more acceptable to security related issues and increase the awareness of security. It is important for leaders to motivate followers to perform in excess of expectations (Yukl & Van Fleet, 1992; Bass, 1985). Other studies also found that top management's order and mandates lead to improved quality performance (Anderson et al., 1995; Flynn et al., 1995). Senior management plays several roles: visionary, transformational leadership has four dimensions: charisma, inspiration, intellectual stimulation, and individualized consideration. Through these four dimensions, leaders create a profound impact on their followers (Yammarino et al, 1997). If the top level management sponsors

and values the ideas of security safeguards, developers would consider building security is expected of them and make efforts to actively implement security practices. Borrowing from the notions of institutional theory, a recent study validated how institutional pressures explain the variations in organizational investments in information security control resources (Cavusoglu et al., 2015). We surmise that top management support can act as a surrogate for institutional pressures thus promoting additional resources to control related activities. If management desires and demands clean and secure code and also provides incentives for application developers, security will be greatly improved.

Security policy

The protection of information systems is a critical problem faced by organizations. Developing security policy and enforcing security policy are essential in protecting information systems. There are many factors affecting implementation of a successful security policy in an organization. Karyda et al. (2005) explore the processes of formulating, implementing and adopting a security policy in two different organizations and propose a theoretical framework based on the theory of contextualism. Each organization has its own characteristics and security policy is context specific. They highlight the dynamic nature of the application of security policies and bring forth contextual factors that affect their successful adoption. The application of a security policy is of utmost importance for managing the security of information systems. We expect that companies enforcing its established security policy will produce applications with high security-quality. Therefore, we test the hypotheses below.

H1: A greater management support to security implementation during the software development process will result in better security policies during the systems development process.

H2: A greater rigor in the specification of security policies will result in higher security quality of the system.

DEVELOPMENT PROCESS CONTROL

Research has shown that the deployment of methodologies by IS developers is primarily associated with a hierarchical culture that is oriented toward security, order, and routinization (Iivari & Huisman, 2007). Application development goes through analysis, design, development and maintenance stages. Effective process control leads to better software quality and process efficiency (Ravichandran & Rai, 2000). Organizations rely on their employees to follow the rules they

establish. Especially in security implementation, such rule adherence is critical to the security quality of the software developed. Formal control and social control individually and interactively enhance both in- and extra-role security behaviors (Hsu, Shih, Hung, & Lowry, 2015). There also have been significant studies dealing with the ability of the organizations to regulate employee conduct in general (Tyler & Blader 2005). Security awareness is a process that aims at changing individuals' perceptions, values, attitudes, behavior, norms, work habits, and organizational culture and structures with regard to secure information practices (Tsohou et al., 2015). Security should be built from the ground up and emphasize throughout application development lifecycle. A life cycle process emphasizing security assurance at each phrase is necessary to improve the overall security of applications (Gilliam et al., 2003). The extent to which the established security standards are being followed in the actual development of information system is an important contributor to the overall security quality.

While there are numerous process control checklists that are widely available, there is a general belief and acceptance from practitioners that these lists alone will not solve the problems of security (Hayes, 2009). Research in this area has generally relied on the deterrence theory to explain user behaviors that are either supportive or disruptive of IS security and the results of this research have been mixed (D'arcy and Herath, 2011). For example, in another study there was insufficient support perceived severity and response cost as being predictors of information systems security policy behavioral compliance intentions (Ifinido, 2012). But, in the context of the present research, it is the software developers' and the end-users' behavior that is relevant to predict the security quality of the product developed, and hence its inclusion in our model will further develop our understanding of how to build security in during systems development.

H3: A greater management support to security implementation during the software development process will result in a better development process control during the systems development process.

H_4: A greater rigor in the control during the development process will result in a higher security quality of the system.

Interactive effect of value congruence and commitment to skills development
Security skills development can help developers to be sensitized to the potential impact of security problems on organizations at large. Due to the rapid development of technologies and increasing dynamics of the business environment, improving security requires training of developers (Nadeem, Allen & Williams, 2015; Jain & Shanbhag, 2012; Bishop, 2003). Educating developers

on the need for good coding practices can result in good coding habits that improve security. At the same time, it is also important to recognize the management commitment to provide security skills development program alone is not sufficient. The moderating effect of situational factors has long been recognized management literature but has not received sufficient attention in IS research. The Influences of individual and situational characteristics on training effectiveness has been proposed (Mathieu et al, 1992). Value Congruence (VC) refers to the congruence of employee's values with those of the company. Here we explore whether there is a relationship between the security goals of the company and the corresponding intrinsic values of an individual employee. We posit that given the same level of training, the more congruence of employee values with the security goals of the company, the better security quality of software products. The interaction of the personal influences on the commitment to develop security skill has not been studied thus far in IS literature. The interactive relationship between goals of a company in providing an employee security skills related training and the employees' value congruence is hypothesized as follows:

H_5: Commitment of organizations to provide security skills development resources and the congruence of employees' personal goals with a company's goal together influence the security quality of the software product.

Legitimacy

Legitimacy refers to an employee's judgments on the appropriateness of the rules. Although the definition of legitimacy can be very complex and varied (Bitektine and Haack 2015, Suchman 1995), for our purposes of explaining security quality, we simply refer to it as the extent to which a participant in the software development environment may perceive the security guidelines to be appropriate or not. Tyler and Blader (2005) explore the two antecedents of rule-following behavior (the other being moral value congruence discussed next) and compare two strategies for achieving rule and policy adherence: (1) an extrinsically oriented command-and-control model and (2) an intrinsically oriented self-regulatory model. Their findings of both studies suggest that the influence of the self-regulatory strategy exceeds that of the command-and-control approach. We do not hypothesize a direct relationship between legitimacy and software quality as legitimacy by itself cannot ensure software quality but only serve to enhance positive employee behavior when parallel security policies are concurrently present. Given the same level of legitimacy, the more adherence to security policy, the better security quality of software products. The literature agrees that the major threat to IS security is constituted by careless employees who do not comply with organizations' IS security policies and procedures. (Pahnila, Siponen,

M, & Mahmood, 2007). Adhering to security policy guidelines is a form of rule-following behavior and hence we propose the following interaction hypothesis of Legitimacy and Security Policy.

H_6: Employees' perception of the legitimacy of security policy guidelines and security policy guidelines together influence the security quality of the software product.

Attitude

Consumer behavior literature has affirmations on how attitude affects behavior (Ajzen & Fishbein, 1977). This attitudes behavior model has been adopted to IT field and studied extensively in Technology Acceptance Model (Davis 1993; Agarwal and Prasad, 1999). Previous research has shown that how certain attitudinal dimensions such as morality can influence certain computer-related behavior (Gattiker & Kelly, 1999). An employee's attitude toward compliance determines intention to comply with the ISP (Bulgurcu et al. 2010). The premise for including attitude in our model is that during application development, developers may have different attitude and value towards security procedures and practices and as the study cited above has posited this may influence the employee compliance behavior. In the context of the adoption of software process innovations, it has been shown that how perceptions of productivity and quality benefits can explain how developers perceive the usefulness of software process innovations which in turn explain some variance in security process improvement (SPI) use. The SPIs must be perceived as useful to a developer for it to be adopted during the software development process (Green, Hevner & Webb Collins, 2005).

It is commonly believed that for effective security, users have to make a conscious decision to comply with the organization's security policies and adopt computer security behavior (Ng, Kankanhalli & Xu 2009). We, therefore, propose the following:

H_7: A positive attitude to enforcing security standards will result in greater security quality of the system.

RESEARCH METHODS

A *survey* is a quantitative method for testing hypotheses/relationship between research constructs (Burrell & Morgan, 1979). As stated in our research objectives, we intend to test factors affecting security quality. Therefore, a survey is an appropriate method for this study.

Measures

Our measures intend to capture how companies develop and enforce secure coding practices, self-assess code during development, implement security checks into the quality assurance cycle and consider security during change control. We adopted prior measures for the constructs in our model. A seven-point Likert scale is used to measure the constructs. Security policy, management support, training, process control, security quality is adapted from Ravichandran and Rai's (2000) general software quality model. Measures for attitude were adapted from Davis' work (1993). Both Legitimacy and Value Congruence were adapted from a study explaining employees rule-following behavior in a general management context (Tyler and Blader 2005). Appendix A identifies the items that make up each of these measures and the source from which they were drawn.

Data Collection

We used two methods to collect data: paper-based and online. The respondents were selected on the basis of their active involvement in systems development or in a managerial capacity such as project managers. We distributed the survey through key personnel in major corporations around the Midwest region in the United States. The link to the online survey was emailed to the respondents. Despite being a form of convenient sampling, this method of distribution ensured that the survey reached key personnel who are qualified to answer security related questions. A total of 300 questionnaires were distributed and we collected a total of 116 responses of which 85 were paper-based and 31 was on online representing a response rate of 39 percent. All respondents are software engineers with only less than 1% of the respondents having experience of less than a year. We performed a t-test on the difference between the paper-based and online survey on age, gender and study constructs. We found no significant differences between the paper-based and online survey on age, gender, attitude, management support, security policy, development control, value congruence, and legitimacy. After data cleansing 114 responses were used for data analysis.

The demographic profile of the respondents was as follows: 75% of the respondents are male and 25% are female. 72.6% of the respondents have over five years of IT experience, 26.5% have one to five years IT experience and 0.9% have IT experience less than a year. Figure 2 below shows the relative age distribution of the respondents.

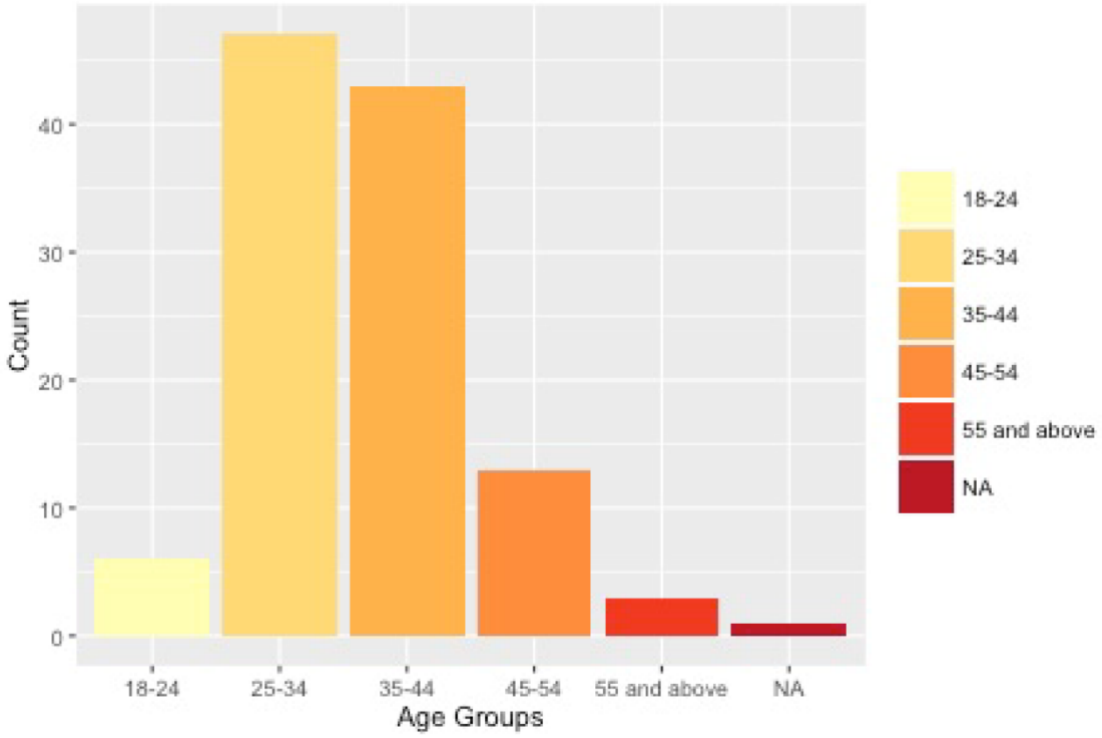

Figure 2: Age Distribution of Survey Participants

ANALYSIS AND RESULTS

We assessed the validity of our survey instrument as discussed below. The measurement model was validated using the established procedure as explained below.

Content validity

All constructs were adapted from prior literature. A previous section on measures highlights the adoption of constructs from previous studies and describes how the measures have content validity.

Convergent validity

Convergent validity refers to that measures of the constructs should be related to each other. PLS-Graph 3.0 was used to analyze the data. Table 1 below shows the

descriptive statistics, composite reliability. The composite reliability for the eight constructs is in the range of 0.82 to 0.95, greater than the suggested 0.7 (Fornell & Larcker, 1981), indicating sufficient internal reliability.

Table 1: Descriptive Statistics

CONSTRUCTS	MEAN	STANDARD DEVIATION	COMPOSITE RELIABILITY
MSUPPORT	2.50	1.30	0.91
ATTITUDE	2.10	1.24	0.95
SPOLICY	2.76	1.29	0.93
SQUALITY	2.41	1.27	0.94
DC	3.00	1.38	0.93
COMMITMENT TO SECURITY	3.35	1.52	0.93
SKILLS DEVELOPMENT			
VALUE	2.21	1.10	0.90
LEGITIMACY	2.41	1.10	0.82

Chin (1998) suggest that convergent validity is established when the loadings are statistically significant and greater than 0.7. As shown in Table 2 all the item loadings have met this rule. Since the items we used are well established in the literature and our data show satisfactory loadings, no items were deleted. In addition to the reliability and loadings, the average variance extracted (AVE) for the eight constructs is between 0.70 and 0.83, higher than the recommended 0.5 (Fornell and Larcker, 1981). We, therefore, conclude that convergent validity for the constructs is established.

Table 2. Loadings DISCRIMINANT VALIDITY

CONSTRUCT	ITEMS	LOADINGS	STANDARD ERROR	T-STATISTIC
SPOLICY	SPOLICY1	0.90	0.02	39.45
	SPOLICY2	0.89	0.04	25.09
	SPOLICY3	0.86	0.03	30.24
	POLICYS4	0.88	0.03	30.79
	SPOLICY5	0.80	0.04	22.90
ATTITUDE	ATTITUDE1	0.88	0.03	30.01
	ATTITUDE2	0.91	0.03	28.72
	ATTITUDE3	0.93	0.03	33.56
	ATTITUDE4	0.85	0.04	19.52
	ATTITUDE5	0.91	0.03	31.09
MSUPPORT	MSUPPORT1	0.88	0.04	21.98
	MSUPPORT2	0.88	0.03	30.34
	MSUPPORT3	0.89	0.03	33.64
SQUAL	QUALITY1	0.88	0.04	22.91
	QUALITY2	0.94	0.01	70.46
	QUALITY3	0.91	0.02	37.79
DC	DC1	0.84	0.05	16.10
	DC2	0.88	0.03	34.36
	DC3	0.89	0.02	40.70
	DC4	0.85	0.05	17.96
	DC5	0.84	0.04	22.51
COMMITMENT TO SECURITY SKILL DEVELOPMENT	COMMIT1	0.88	0.03	28.24
	COMMIT2	0.92	0.02	56.08
	COMMIT3	0.87	0.05	18.81
	COMMIT4	0.84	0.04	19.83
LEGITIMACY	LEGITIMACY1	0.74	0.07	10.50
	LEGITIMACY2	0.72	0.08	8.77
	LEGITIMACY3	0.73	0.08	9.11
	LEGITIMACY4	0.74	0.08	9.46
VALUE CONGRUENCE	VALUE1	0.83	0.05	18.21
	VALUE2	0.82	0.06	13.54
	VALUE3	0.93	0.02	43.14

Discriminant validity means that measures of the constructs should not be related to each other. Campbell and Fiske (1959) suggest that item to construct correlation can be used to establish discriminant validity. Items should be correlated higher to their corresponding construct than to other constructs. Table 3 shows the correlations between constructs and the bold number on the diagonal is the square root of average variance extracted. The square root of the average variance extracted from a given construct is higher than its correlation with other constructs.

Table 3. Correlation Matrix and Square Root of Average Variance Extracted (AVE)

		1	2	3	4	5	6	7
SECURITY POLICY	1	0.87						
ATTITUDE	2	0.32	0.89					
MANAGEMENT SUPPORT	3	0.56	0.11	0.88				
SECURITY QUALITY	4	0.63	0.37	0.37	0.91			
DEVELOPMENT PROCESS CONTROL	5	0.72	0.22	0.58	0.60	0.86		
VALUE * COMMITMENT TO SECURITY SKILL DEV.	6	0.64	0.18	0.52	0.60	0.72	0.80	
LEGITIMACY * SECURITY POLICY	7	0.80	0.27	0.54	0.61	0.62	0.67	0.66

In addition, we also produce a table show item-construct correlation. As shown in Table 4, all items correlate higher with their intended construct than with other constructs. Thus, we conclude that discriminant validity is established.

Table 4. Item-to-Construct Correlation

	MSUPPORT	SPOLICY	COMMIT	DC	SQUAL	ATTITUDE	LEGIT	VALUE
MSUPPORT1	.88	.43	.54	.45	.23	.07	.31	.12
MSUPPORT2	.90	.49	.58	.56	.32	-.03	.37	.14
MSUPPORT3	.89	.59	.52	.57	.43	.25	.35	.19
SPOLICY1	.18	.90	.53	.63	.57	.32	.29	.27
SPOLICY2	.23	.89	.51	.62	.52	.34	.27	.23
SPOLICY3	.18	.87	.48	.61	.51	.23	.40	.30
SPOLICY4	.00	.88	.51	.63	.59	.25	.32	.26
SPOLICY5	-.03	.81	.56	.65	.56	.24	.37	.33
SPOLICY6	-.10	.56	.43	.47	.42	.10	.40	.20

COMMIT1	.16	.52	.88	.62	.40	.11	.35	.31
COMMIT2	.14	.58	.93	.74	.46	.19	.34	.32
COMMIT3	.30	.48	.88	.65	.33	.14	.34	.25
COMMIT4	.10	.51	.84	.62	.32	.06	.23	.24
DC1	.03	.55	.54	.84	.50	.16	.35	.19
DC2	.04	.70	.61	.88	.58	.30	.30	.21
DC3	-.02	.60	.69	.89	.52	.26	.35	.28
DC4	.08	.66	.68	.85	.47	.09	.29	.26
DC5	.22	.58	.70	.85	.50	.13	.27	.27
SQUAL1	-.03	.51	.34	.50	.89	.42	.39	.29
SQUAL2	.00	.62	.40	.60	.96	.32	.37	.33
SQUAL3	.08	.62	.47	.58	.92	.30	.48	.41
ATTITUDE1	-.02	.30	.13	.28	.41	.88	.11	-.02
ATTITUDE2	.04	.29	.12	.20	.26	.91	.09	-.04
ATTITUDE3	.14	.30	.13	.17	.34	.93	.21	-.03
ATTITUDE4	.00	.24	.12	.12	.28	.85	.12	.00
ATTITUDE5	.09	.28	.15	.18	.34	.91	.23	.01
LEGITIMACY1	.18	.21	.31	.28	.35	.03	.74	.28
LEGITIMACY2	.12	.25	.14	.20	.23	.19	.72	.13
LEGITIMACY3	.10	.25	.22	.20	.27	.14	.73	.27
LEGITIMACY4	.10	.38	.34	.34	.45	.19	.75	.26
VALUE1	.22	.53	.48	.47	.49	.23	.46	.83
VALUE2	.15	.43	.46	.43	.46	.25	.53	.82
VALUE3	.18	.60	.61	.62	.67	.23	.48	.93

Structural Model

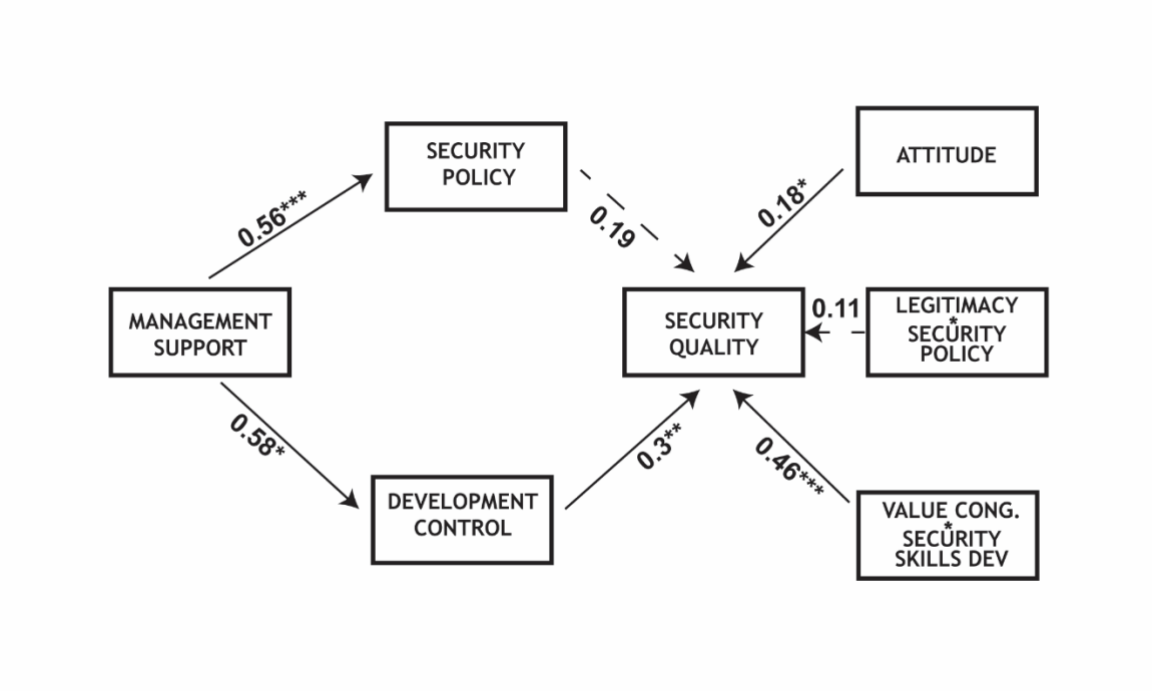

Figure 3. Path Model (significance level of t statistics: *=0.1; **=0.5; ***=0.01)

Figure 3 shows the path model, path coefficients, and the associated significance level. The solid lines indicate a significant relationship between the constructs. The dashed lines show the insignificant relationships. The path coefficient is 0.56 for the path between management support and security policy and it is significant at 0.01 level with an R square of 31.6%. The path coefficient is 0.58 for the path between management support and development control and it is significant at 0.01 level with an R square of 33.7%. When five constructs (security policy, development control, attitude, legitimacy*security, and value congruence*training) are used as independent variables to predict security quality, three paths are significant at 0.05 level or above. Collectively, these five constructs explain 55% of the variance in security quality. The path coefficients for the paths (development control→security quality, and value congruence * commitment to security skills development→security quality) are 0.3, and 0.46 respectively. The path coefficient is 0.18 between attitude and security quality,

and it is significant at 0.10 level. Two paths are not significant (security policy→security quality, legitimacy*security policy→security quality).

DISCUSSION

Our data and analysis support the claim that management support is positively related to security policy and development control. Management is the champion and with their support, security policies are developed to meet the expectations and goals set by the management. In terms of development control, management plays a key role too. Developers are more likely to incorporate security features into their program if they sense the strong emphasis from the management. If management stresses the importance of security training and allocates resources for it, the overall awareness of security will be improved if employees have congruent values with the security goals of the company.

The insignificant relationship between security policy and security quality indicate that though companies have security policies, that does not necessarily lead to high security-quality. Our results suggest that having security policy alone may not be sufficient to have high security-quality. Unless accompanied by a mechanism to ensure that developers follow established security policies, translating policies into actual security quality may not be possible. Recognizing this importance of compliance, a recent study has developed and advocated training programs to ensure compliance with security guidelines (Puhakainen & Siponen, 2010). In practice, there may be workarounds for security policies. Although the literature on security policy compliance suggests that legitimacy moderates the relationship between security policy and security quality (Tyler & Blader, 2005), our data do not support this. Conceptually legitimacy items focus on policy and guidelines, it is reasonable to refer the legitimacy and security policy are moving in the same direction when predicting security quality. It is not a complete surprise to see this interaction relationship not confirmed. The strong relationship between development control and security quality, and between the interaction (training x value) and security quality show that these are the importance of development control, training, and value*training in security quality.

Theoretical & managerial implications

The current model incorporates two aspects of individual traits that were previously studied as antecedents of rule-adherence behavior (Tyler and Blader 2005). Incorporating individual traits into a set of organizational contextual variables allows us to further explore relationships explaining why it is not enough just to study what an organization does to improve its security awareness.

We found a significant relationship between value congruence and commitment to skill development (training) that can improve security quality of an organization's software systems. These findings highlight the need to recruit and retain software development personnel who have the right attitude toward security awareness and values that reflect the usefulness of acquiring additional skills through training.

Our results first show the importance of managerial support to foster a climate to establish security policies. Systems development process can have sufficient control to incorporate sufficient security related featured into the products developed and finally to display organizational commitment to provide to systems development personnel necessary security skills through training along with strategies to develop employees' value congruence with the security goals and availability of training resources. These results offer two important prescriptive guidelines to improve the security of the software product developed: allocating resources to development control and an improved commitment to security skill development through a training of the developers are the right approach to enhance security quality simultaneously employing strategies to improve development personnel's value congruence with security skills development. These strategies could include programs to improve security awareness among developers.

LIMITATIONS AND FUTURE STUDY

We used the convenience sampling procedure to test our research model. Despite its usefulness in reaching the right target audience, this sampling method may be viewed by some as a limitation. In terms of generalizability, our results may be generalized with caution. Our finding of the absence of a significant relationship between security policy and security quality will have to be further explored. Although the importance of fostering rule compliance behavior has been identified, the exact mechanisms by which this can occur needs to be studied further. Our additional finding of an insignificant interaction between legitimacy and security policy should also be viewed with some caution; there is a strong natural appeal to completely ignore this interaction in future studies. Despite the existence of security policies, there seem to be both additional individual traits that make organizational infrastructure security adherence more successful. It is our belief that future studies extend the current model to more comprehensively understand what really causes compliance with security guidelines in an organization.

CONCLUSION

This study addresses concerns of lack of security in software products at different levels. It highlights the need to focus on executive leadership, project structure, and process areas by fostering a secure application development culture. This multilevel approach to analyzing security awareness has been advocated before. For example, Tsohu et al (2015) argue that any analysis done solely at the organizational level or an individual level will provide only a distorted view. In the current model, we have included considerations of individual level and brought two constructs – Legitimacy and value congruence – that are generally thought to be an antecedent of compliance behavior (Tyler & Blader, 2005).

An appropriate curriculum to improve security awareness is recommended as there is a general consensus that training on security awareness should start from the college level. There exists a gap between the emphasis on teaching security during program development and its need in the workplace. Despite this deficiency, universities are becoming increasingly involved in developing security-related courses. To some extent, security can be embedded into every course in the information systems area (Goodwin, 2003). Undergraduate curriculum should cover such security contents as analyzing the security of code, model threats and vulnerabilities, fix programs, and the differences between secure and insecure programming languages. By doing so, after students join the labor force, they are better trained with security concepts, procedures, coding practices.

REFERENCES

Adams, A., and Blandford, A. (2005), "Bridging the Gap Between Organizational And User Perspectives Of Security In The Clinical Domain," International Journal of Human-Computer Studies, 63(2): 175-202.

Agarwal, R., and Prasad, J. (1999), "Are Individual Differences Germane To The Acceptance Of New Information Technologies?" Decision Sciences, 30(2):361-391.

Ajzen, I., and Fishbein, M. (1977), "Attitude-Behavior Relations: A Theoretical Analysis And Review Of Empirical Research," Psychological Bulletin, 84(5): 888.

Allen, M.P., and Panian, S. (1982), "Power, Performance And Succession In The Large Corporation," Administrative Science Quarterly, 27(4): 538-547.

Anderson, J. C., Rungtusanthanam, M., Schroeder, R., and Devaraj, S. (1995), "A path analytic model of a theory of quality management underlying the Deming management method: Preliminary empirical findings," *Decision Sciences*, *26*(5): 637-658.

Barragáns-Martínez, B., Costa-Montenegro, E., & Juncal-Martínez, J. (2015). Developing a recommender system in a consumer electronic device. *Expert Systems with Applications*, *42*(9), 4216-4228.

Bass, B. M. (1985), *Leadership and performance beyond expectations*, Free Press, New York.

Bergaval-Kåreborn, B., and Howcroft, D. (2014), "Persistent Problems and Practices In Information Systems Development: A Study Of Mobile Applications Development And Distribution," *Information Systems Journal*, *24*(5): 425-444.

Bishop, M. (2003), "What is computer security?" *Security & Privacy IEEE*, *1*(1): 67-69.

Bitektine, A., and Haack, P. (2015), "The "macro" and the "micro" of legitimacy: Toward a multilevel theory of the legitimacy process," *Academy of Management Review*, *40*(1): 49-75.

Boeker, W. (1992), "Power and managerial dismissal: Scapegoating at the top," *Administrative Science Quarterly*, *37*(3): 400-421.

Bulgurcu, B., Cavusoglu, H., and Benbasat, I. (2010), "Information security policy compliance: an empirical study of rationality-based beliefs and information security awareness," *MIS quarterly*, *34*(3): 523-548.

Burrell, G., and Morgan, G. (1979), Assumptions about the nature of science. In Sociological Paradigms and Organizational Analysis 1979, Portsmouth, NH: Heinemann, 1–35.

Čeke, D., and Milašinović, B. (2015), "Early Effort Estimation in Web Application Development," *Journal of Systems & Software*. 103: 219-237.

Cambell, D. T., and Fiske, D. W. (1959), "Convergent and Discriminant Validation by The Multitrait-Multimethod Matrix," *Psychological Bulletin* 56: 81-105.

Cavusoglu, H., Cavusoglu, H., Son, J. Y., and Benbasat, I. (2015), "Institutional Pressures In Security Management: Direct And Indirect Influences On

Organizational Investment In Information Security Control Resources," *Information & Management, 52*(4): 385-400.

Chin, W. W. (1998), The partial least squares approach to structural equation modeling. In Modern Methods for Research (G. A. Marcoulides, Ed.), Mahwah, NJ: Lawrence Erlbaum, 295-336.

D'arcy, J., and Herath, T. (2011), "A Review and Analysis Of Deterrence Theory In The IS Security Literature: Making Sense Of The Disparate Findings," *European Journal of Information Systems, 20*(6): 643-658.

Davis, F. D. (1993), "User Acceptance of Information Technology: System Characteristics, User Perceptions And Behavioral Impacts," *International Journal of Man-Machine Studies*, 38(3): 475-487.

Deming, E. (1986), Out of the crisis, MIT Center for Advanced Engineering, Cambridge.

Dhillon, G., and Backhouse, J. (2000). "Information system security management in the new millennium," *Communications of the ACM, 43*(7): 125-128.

Denning, D. E. (2015), "Toward more secure software," *Communications of the ACM, 58*(4): 24-26.

Fornell, C., and Larcker, D. F. (1981), "Evaluating Structural Equation Models With Unobservable Variables And Measurement Error," *Journal of Marketing Research, 18*(1): 39-50.

Flynn, B., Schroeder, R. G., & Sakaibara, S. (1995), "The Impact Of Quality Management Practices On Performance And Competitive Advantage," *Decision Sciences*, 26(5): 659-692.

Gattiker, U. E., and Kelly, H. (1999), "Morality and computers: Attitudes and differences in moral judgments," *Information Systems Research, 10*(3): 233.

Giliam, D. P., Wolfe, T. L., Sherif, JS, and Bishop, M. (2003). Software security checklist for the software life cycle. Paper presented at the *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003. WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on,* 243-248.

Goodwin, B. (2003), "Secure Coding Course," *Computer Weekly*, 3.

Green, G. C., Hevner, A. R., and Webb, C. R. (2005), "The Impacts Of Quality And Productivity Perceptions On The Use Of Software Process Improvement Innovations," *Information and Software Technology, 47*(8): 543-553.

Hayes, F. (2009), "More than a list," *Computerworld*, 43(3): 40-40.

Howard, M., & Lipner, S. (2006), The security development lifecycle: SDL, a process for developing demonstrably more secure software. Redmond, Wash: Microsoft Press.

Hsu, J. S. C., Shih, S. P., Hung, Y. W., and Lowry, P. B. (2015), "The Role of Extra-Role Behaviors and Social Controls in Information Security Policy Effectiveness," *Information Systems Research.* 26(2): 282-300.

Ifinedo, P. (2012), "Understanding information systems security policy compliance: An integration of the theory of planned behavior and the protection motivation theory," Computers & Security, 31(1): 83-95.

Iivari, J., & Huisman, M. (2007), "The relationship between organizational culture and the deployment of systems development methodologies," MIS Quarterly, 31(1): 35-58.

Jain, A. K., & Shanbag, D. (2012), "Addressing Security And Privacy Risks In Mobile Applications," *IT Professional* (5): 28-33.

Jones, R. L., and Rastogi, A. (2004), "Secure coding: Building security into the software development life cycle," Information Security Journal: A Global Perspective, 13(5): 29-39.

Karyda, M., Kiountouzis, E., and Kokolakis, S. (2005), "Information Systems Security Policies: A Contextual Perspective," Computers & Security, 24(3): 246.

Kayworth, T., and Whitten, D. (2010). "Effective Information Security Requires A Balance Of Social And Technology Factors," MIS Quarterly Executive, 9(3): 2012-52.

King, S. (2004), Bridging the gap between security and developers, Computer Weekly 32.

Landwehr, C. E., Bull, A. R., McDermott, J. P., and Choi, W. S. (1994), "A Taxonomy Of Computer Program Security Flaws," ACM Computing Surveys (CSUR), 26(3): 211-254.

Lipner, S. B. (2015), "Security Assurance," Communications of the ACM, 58(11): 24-26.

Lu, J., Wu, D., Mao, M., Wang, W., and Zhang, G. (2015), "Recommender System Application Developments: A Survey," Decision Support Systems, 74: 12-32.

Mathieu, J. E., Tannenbaum SI, and Salas, E. (1992), "Influences of Individual And Situational Characteristics On Measures Of Training Effectiveness," Academy of Management Journal, 35(4): 828-847.

NABI, F. (2005), "Secure business application logic for e-commerce systems," Computers & Security, 24(3): 208.

Nadeem, M., Allen, E. B., and Williams, B. J. (2015), A Method for Recommending Computer-Security Training for Software Developers: Leveraging the Power of Static Analysis Techniques and Vulnerability Repositories. In Information Technology-New Generations (ITNG), 2015 12th International Conference on (pp. 534-539). IEEE.

Ng, B., Kankanhalli, A., and Xu, Y. (2009), "Studying users' computer security behavior: A health belief perspective," Decision Support Systems, 46(4): 815-825.

Pahnila, S., Siponen, M., and Mahmood, A. (2007, January) Employees' behavior towards IS security policy compliance. In System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on (pp. 156b-156b). IEEE.

Popa, M. (2009), "Detection of the Security Vulnerabilities in Web Applications," Informatica Economica, 13 (1): 127-136.

Puhakinen, P., and Siponen, M. (2010), "Improving Employees' Compliance through Information Systems Security Training: An Action Research Study," MIS Quarterly, 34(4): 757-778.

Ravichandran, T., and Rai, A. (2000). "Quality management in system development: An organizational system perspective," MIS Quarterly, 24(3): 381- 415.

Ravichandran, T., and Rai, A. (2003). "Structural analysis of the impact of knowledge creation and knowledge embedding on software process capability," IEEE Transactions on Engineering Management, 50(3): 270-284.

Ryoo, J., Kazman, R., and Anand, P. (2015), "Architectural Analysis for Security," IEEE Security & Privacy (6): 52-59.

Schonberger, R. J., & Ansari, A. (1984). "Just-In-Time" Purchasing Can Improve Quality. *Journal of Supply Chain Management*, *20*(1), 2-7.

Schumacher, M., Fernandez-Buglioni, E., Hybertson, D., Buschmann, F., and Sommerlad, P. (2013), Security Patterns: Integrating security and systems engineering. New York City, NY: John Wiley & Sons.

Scott, D., & Sharp, R. (2002, May). Abstracting application-level web security. In *Proceedings of the 11th international conference on World Wide Web* (pp. 396-407). ACM.

Sood, S. K. (2012), "A combined approach to ensure data security in cloud computing," Journal of Network and Computer Applications 35(6): 1831-1838.

Suchman, M. C. (1995), "Managing legitimacy: Strategic and institutional approaches," Academy of Management Review, 20(3): 571-610.

Tsohou, A., Karyda, M., Kokolakis, S., and Kiountouzis, E. (2015), "Managing the introduction of information security awareness programmes in organisations," European Journal of Information Systems 24(1): 38-58.

Tyler, T. R., and Blader, S. L. (2005), "Can businesses effectively regulate employee conduct? The antecedents of rule following in work settings," Academy of Management Journal 48(6): 1143-1158.

Van Wyk, K. R., and Steven, J. (2006), "Essential Factors For Successful Software Security Awareness Training," Security & Privacy, IEEE, 4(5): 80-83.

Viega, J., & McGRAW, G. (2001), Building secure software: how to avoid security problems the right way. Pearson Education.

Wang, H., and Wang, C. (2003), "Taxonomy of security considerations and software quality," Communications of the ACM, 46(6): 75-78.

WorkmaN, M., Bommer, W. H., and Straub, D. (2008), "Security lapses and the omission of information security measures: A threat control model and empirical test," Computers in Human Behavior 24(6): 2799-2816.

Yammarino, F. J., Dubinsky, A. J., Comer, L. B., & Jolson, M. A. (1997). Women and transformational and contingent reward leadership: A multiple-levels-of-analysis perspective. *Academy of Management Journal*, *40*(1), 205-222.

Yukl, G., & Van Fleet, D. (1992). Theory and Research on Leadership in Organizations.-Palo Alto.

APPENDIX A

Measures

IS management's support for security (Ravichandran & Rai, 2000)

1. IS chief executive assumes responsibility for security.
2. IS chief executive is evaluated for security performance.
3. IS chief executive supports security quality improvement process.

Security policy (Ravichandran & Rai, 2000)

1. IS management has clear security quality and goals.
2. Quality goals relating to security are very specific.
3. Significant importance is attached to security quality in relation to cost and schedule objectives.
4. There is a comprehensive security quality plan
5. Security policy guidelines are understood by the project team.
6. For this project, security requirements were developed along with functional requirements

Commitment to security skill development/Training (Ravichandran & Rai, 2000)

1. Training in security management tools and techniques are given to project professionals.
2. Security skill training is given to IS personnel.
3. Resources are made available for training project personnel.
4. Security awareness programs are available to project professionals.

Development process control (Ravichandran & Rai, 2000)

1. Security Performance standards have been established for design.
2. Security Performance standards have been established for programming.
3. Security Performance standards have been established for testing.
4. Security guidelines are revised periodically
5. Security metrics are used for evaluating system security.

Security quality (Ravichandran & Rai, 2000)

1. Users perceived that the system meet the intended security requirements.
2. Users are satisfied with the overall security of the system.
3. Users have no complaints on the security aspects of the system.

Attitude (Davis, 1993)

1. Enforcing security during development is good.
2. Enforcing security during development is useful.
3. Enforcing security during development is beneficial to the company.
4. Enforcing security during development is beneficial to me.
5. Enforcing security during development is valuable.

Legitimacy (Tyler & Blader 2005)

1. A project professional should accept the policies spelled out by the organization even when they may be perceived as being wrong.
2. Deviating from the security policies is seldom justified.
3. Someone who disregards the security policies hurts their work group and the security quality of the project.
4. Projects are most successful when employees follow project guidelines.

Value Congruence (Tyler & Blader 2005)

1. I find that my values on security and the values where I work are very similar.
2. What my company stands for in defining the security goals is important to me.
3. I agree with the values that define the security goals of this project.