ABSTRACT

This paper provides a framework to reconcile the apparent duality between Information Systems Development and Information Systems Security Development. Though there is a substantial foundation of literature on approaches and methodologies for Information Systems Development as well as for Information systems Security Development, the two are not well integrated. Part of the cause of the continuing problems with information systems security is the often ad hoc security implementation. This "security after the fact" can lead to incompatibility between the system and the security of the system. It appears that a theoretically grounded and methodological approach for integrating security with Information Systems Development has been lacking. This paper proposes a basic framework as a step toward such an approach.

1. INTRODUCTION

Information Systems (IS) Security issues continue to plague organizations (Dhillon 2001) despite steadfast efforts at controlling the problem. According to the 2005 CSI/FBI Computer Crime and Security Survey, total monetary losses due to IS Security breaches topped 130 million dollars (Gordon, Loeb, Lucyshyn, and Richardson 2005). Over 56% of organizations surveyed reported unauthorized use of computer systems within the last 12 months (Gordon et al. 2004). A major contributing factor to this ongoing problem may be that IS Security tends to be an afterthought with systems development. This phenomenon, described by Baskerville (1993) as ad hoc IS Security implementation, can lead to incompatibility between the system and the security of the system. It appears that a theoretically grounded and methodological approach is lacking for integrating IS Security with information systems development (ISD). In this paper, we propose an integration method using the Socio-Technical Design (STD) (Lyytinen, Mathiassen, and Ropponen 1998) approach.

Both Baskerville and Siponen describe the evolution of IS Security development. The first three generations are identified as checklists, mechanistic engineering methods, and logical transformation methods, respectively (Baskerville 1993). Siponen (2001) extends this with a fourth generation that accounts for Dhillon and Backhouse’s (2000) responsibility modeling. Both Baskerville (1992) and Siponen (2001) recognize that a
developmental duality exists between IS Security development and ISD. It is this very duality that may be at the root of many of the problems related to IS security.

In examining this developmental duality, the remainder of this paper is structured as follows: First, in section 2, a review of how other researchers have approached the problem is given. Then, section 3 moves into a discussion regarding ISD methodologies. Section 4 provides an overview of the underlying theoretical framework, the Socio-Technical Model, to justify the proposed method for IS Security integration, which is discussed in section 5. Section 6 shows an example of IS Security integration with a selected ISD methodology. The paper concludes by summarizing the findings and identifying possible future research in the area.

2. PREVIOUS RELATED WORK

Much of the work examined for this paper, though rigorous and well thought out in the ISD and IS Security integration attempts, is nevertheless limited in scope with respect to the security side. Inmore, Esichaikul, and Batanov (2003) present an IS Security-oriented extension to the object model. They focus on the analysis phase of the Object Oriented Analysis and Design (OOAD) ISD methodology but make the argument that applying and integrating stringent security at this level will permeate the entire process. They do this by changing the very structure of the object class itself to include a security extension.

The major drawback of Inmore et al.’s (2003) proposal is that it draws on no literature regarding the security engineering approach chosen. The paper begins with the premise that all of security amounts to Confidentiality, Integrity, and Availability (CIA). By relying solely on CIA, the authors are stuck in what Baskerville (1993) calls the third generation of security development. Soft, or behavioral, issues (Dhillon 2000; Siponen 2001) are not taken into consideration. This potentially could lead to a system which is technically very secure but none-the-less completely vulnerable to internal threats.

Jürjens, Popp, and Wimmel (2003) took a similar approach to Inmore et al. (2003) by creating a security extension to a particular facet of OOAD and assuming it will permeate the entire methodological process. Instead of extending the definition of the actual class as Inmore et al. (2003) did, they propose a method by which one can express security-related information within the diagrams in a UML system specification. They do this by providing four new stereotypes with descriptive tags, namely secrecy, integrity, high, and critical, and describe various levels and types of security. Unfortunately, the approach of Jürjens et al. (2003) suffers from the same problems as that of Inmore et al. (2003). Their view of security focuses solely on CIA. Despite their solid choice in methodologies, the lacking security foundation may lead to vulnerabilities.

Other research that has attempted to produce an integration method but is limited to the technical aspects of security (CIA) include Jones and Rastogi (2004), Breu, Burger, Hafner, and Popp (2004), and Mouratidis, Giorgini, and Manson (2003). On the other
hand, some research in the area has grounded itself in a security foundation (such as the Systems Security Engineering Capability Maturity Model (SSE-CMM)) in order to give itself a solid foundation for the security side of the integration attempt. Lee, Lee, and Lee (2002) and Chan and Kwok (2001) are two examples of such attempts.

Lee et al. (2002) propose an integration model that intertwines all the process activities and deliverables of a Systems Development Lifecycle (SDLC) with Security Engineering (SE) activities. Though based in the SSE-CMM (and several other security models), Lee et al. (2002) hand-picked the SE components they thought were the most important to include in their integration model. Though Lee et al. (2003) validated their model post facto with nine experts, the method of integration used by them seems to be somewhat arbitrary. It was hand crafted and allows for the incorporation of organizational and supporting processes.

Chan and Kwok (2001) performed a similar analysis using the SSE-CMM, but attempted the integration with an OOAD methodology. Interestingly, like Lee et al. (2002), they hand-picked what portions of the SSE-CMM should be integrated without relying on any underlying theoretical framework.

3. ISD METHODOLOGY

Besides the techno-centric security development focus (CIA) and the lack of a solid theoretical grounding of the proposed integration methods, a prevailing trend in the previously discussed literature is a lack of discussion on the justification for the choice of the ISD methodology. Ulrich (2003) points out this lack of critical thinking that often accompanies the decision making process of ISD methodology choice. This section of our paper provides a brief overview of ISD methodologies and justifies our choice of methodology for integration. The section is driven by Iivari, Hirschheim, and Klein’s (2001) philosophical approach to ISD methodology classification. Iivari et al. (2001) begin their classification at an ontological and epistemological level, but we focus more on their lowest level of their classification: methodologies.

An appropriate starting point might be the exhaustive list of methodologies provided by Iivari et al. (2001). From their own list, Iivari et al. (2001) chose 11 methodologies which are representative of all four paradigms they explore. While admirable from a philosophical perspective, their approach is not very pragmatic. For example, one of the four paradigms is radical structuralism, containing a single methodology: “trade unionist.” Outside of their article, it may be difficult to find any reference to this methodology in the IS litterature. A second paradigm included is neo-humanism, also containing only one available methodology: speech-act based information analysis methodology with computer-aided tools. Like the trade unionist approach, this is not a widely used or much cited methodology. Hence, these methodologies shall be eliminated from the analysis presented in our paper.
The two remaining paradigms that Livari et al. (2001) provide are social relativism and functionalism. The social relativist paradigm seems to be relatively obscure, at least in the United States, but it does have a noted methodology, namely the Soft Systems Methodology (SSM) (Checkland, and Scholes 1990). The functionalist paradigm, on the other hand has been a dominant ISD paradigm in the U.S. There are several methodologies within this paradigm that stand out, based on the great number of citations in the major IS journals and their frequent use in practice. These are Structured Analysis and Design (SAD) (Randell, 1969), OOAD (Booch, Rumbaugh, and Jacobson, 1999), and ETHICS (Hirschheim and Klein, 1994).

Because it is based in the same socio-technical epistemology as the upcoming theoretical framework, it is tempting to use Hirschheim et al.’s (1994) ETHICS methodology. It is also tempting to use Checkland et al.’s (1994) SSM methodology due to its heavy focus on soft, or behavioral, issues. Both of these methodologies will be rejected though because of their relative obscurity in the U.S. practitioner world. Between the remaining two methodologies, SAD and OOAD, OOAD appears to be the better choice. This is because OOAD is the emerging dominant ISD methodology that is consistently making inroads to the mature SAD paradigm (George, Batra, Valacich, and Hoffer 2004).

4. SOCIO-TECHNICAL THEORY

The notion of socio-technical system emerged from the labor studies conducted by the Tavistock institute in British coal mining industry in the 1950s. The concept of socio-technical system emphasizes the inter-relationship between humans and technology in an organization. The focus is to enhance efficiency without ignoring human work or social conditions. It is a general approach to the analysis and design of organizational structures. The major sources of influences on the socio-technical perspective have been the concept of socio-technical system, research on small groups, system theory, and principles of job design.

An open socio-technical systems framework, as influenced by system theory, considers work operations as (Badham et al. 2000):

- Systems with interdependent parts.
- Open systems adapting to and pursuing goals in external environment.
- Open socio-technical systems possessing an internal environment made up of separate but interdependent technical and social sub-systems.
- Open socio-technical systems with equifinality, that is, in which system goals can be achieved by different means.
- Open socio-technical systems in which performance depends on jointly-optimizing the technical and social sub-systems.

The socio-technical design principles can be used to guide the individual jobs, technology, work processes and organizational structure. Cherns (1987) advocates a comprehensive set of these principles. The first principle of compatibility means that the process of design should be compatible with the design objectives. The second principle of minimal critical specification implies that the means of achieving the objectives should
be specified, while the objectives should be stated. The third principle refers to variance control, where the variances should be controlled at the source. The fourth principle is the boundary control. The boundaries should not be drawn so as to impede sharing of information, knowledge or learning. The principle of information flow requires that information should be provided to those who require it when they need it. According to the principle of power and authority, people should have access to resources and authority to command them in order to carry out their responsibilities. The multifunctional principle deals with the multiple roles of individuals and teams to increase their response repertoires. The principle of support congruence implies the need of congruency between supporting systems and sub-systems. According to the principle of transitional organization, periods of transition require planning and design, and transitional organizations may be different from the old and the new systems, and are themselves subject to socio-technical design. Finally, the principle of incompletion implies that redesign is continuous and is the function of self-regulating teams.

**4.1. SOCIO-TECHNICAL MODEL FOR SECURITY CONSIDERATIONS**

For the purposes of this paper, we adopt socio-technical model of systems development of Lyytinen *et al.* (1998) to analyze the IS security issues. This model was adapted from Leavitt’s (1964) open system model of organizational change. Leavitt’s model viewed organizations as comprised of four interacting components—task, structure, actor, and technology. Lyytinen *et al.* translated these components into elements of systems development. The socio-technical model of systems development is presented in figure 1. The following discussion is based upon Lyytinen *et al.*’s explanation of the socio-technical model.

The actor component of the model includes various stakeholders in an organization including users, managers, developers and designers. The structure component involves systems of communication, authority and work flow. It includes both the normative (values, norms) and the behavioral dimension. The technology involves different types of tools, methods, hardware and software platforms utilized to develop and implement a system. The task component signifies expected outcomes in terms of goals and deliverables.

An important consideration in this model is that all the four components are related to each other. As such, any changes in one component would have an effect on the others as well. Consequently, any unwarranted condition or state at one component would have an adverse affect on the other components as well as on the entire system. Therefore, the goal is to control any adverse variations in the system and maintain the systems in balance. In the light of the above, the interdependencies between actors, structure, technology, and task assume an important role.
Actor-Structure interdependencies focus on interactions between the structure and the actors. As per Lyytinnen et al. typical concerns are: incentive schemes and sanctions, values and beliefs, and how actors’ behaviors are in concordance with the prevailing organizational structure. Actor-Technology interdependencies deal with the variations created by the misalignment between people and technology. This might arise from implementing untested technologies or mismatching people with inappropriate technology.

Technology-Structure interdependencies address the interactions between technology and the organizational structure. The variations arise as a result of conflict and disparity between the implemented technologies and the existing structures. Task-Technology interdependencies focus on the technological fit with the task. Task-Structure interdependencies deal with the interactions between task and organizational structure. Structures should be aligned with the organizational goals. A misalignment between the two would lead to inefficient outcomes.

Based on our understanding of the socio-technical model we derive a security framework as presented in Table 1. This framework accounts for the techno-centric security aspects (CIA) covered by much of the previous integration literature, but also includes the soft issues that is lacking in many of these previous approaches. These include responsibility, integrity, trust, and ethicality (Dhillon, et al. 2000), and culture, norms, and beliefs (Backhouse, et al. 1996). Furthermore, it also includes the critical backbone of any secure environment, the security policy (Baskerville, et al. 2002). This holistic view of IS Security, grounded in the socio-technical model may provide the foundation for any security integration efforts.

![Figure 1 A Socio-Technical Model of Systems Development](image-url)
<table>
<thead>
<tr>
<th>Components</th>
<th>IS Security Issues</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors</td>
<td>Responsibility, Integrity, Trust, Ethicality</td>
<td>Dhillon &amp; Backhouse (2000)</td>
</tr>
<tr>
<td>Structure</td>
<td>Culture, Norms, Beliefs</td>
<td>Backhouse &amp; Dhillon (1996)</td>
</tr>
<tr>
<td>Technology</td>
<td>Confidentiality, Integrity, Availability, Non-repudiation</td>
<td>Howard (1995)</td>
</tr>
</tbody>
</table>

*Table 1 IS Security STD framework*

5. USE OF THE FRAMEWORK FOR ISD-SECURITY METHODOLOGICAL INTEGRATION

As per Lyytinen et al., “a change in any socio-technical component or relation in a systems development process can create variations which, in the extreme, can lead to a failure of the system development (system), otherwise known as a loss.” The argument promoted in this research is that IS security should be addressed in terms of the four components of the socio-technical model and their interdependencies. We suggest that this leads to an efficient and effective integration of IS Security in the information systems development approaches. The IS Security issues need to be derived from an applicable theoretical basis in the IS security research literature. In order to minimize the loss or avoid system failure, these security concerns should be adapted to integrate with the socio-technical model of system development.

The Socio-Technical Design approach is adopted in this research as it concentrates on both the technical and social sub-systems of an organization. This takes into account not only the formal aspects of an organization but informal aspects like norms and culture as well. Any negative variation in either of these sub-systems or their interactions would lead to an adverse impact on the effectiveness of the organization. Further, the socio-technical model takes into account the organizational structures and reminds us of the importance of the alignment with business objectives. The core focus of the socio-technical model is on the working organization. This objective fits the concerns raised in this paper, since the goal of IS security is ensure a functioning, i.e. working organization.

To show how the proposed IS Security STD framework can be incorporated in a given systems development methodology, an extra column needs to be added to Table 1. This column should contain the discrete components that make up a given ISD methodology. To do this in a rigorous fashion, the appropriate placement of these components must be justified and verified in an iterative process.
6. IMPLEMENTATION OF THE INTEGRATION METHOD

In this paper, we present a first iteration of ISD-Security integration, grounded in the STD theoretical framework using OOAD. According to George et al. (2004), OOAD is driven by the Rational Unified Process (RUP). RUP’s phases of development are *inception, elaboration, construction, and transition*. RUP is an iterative and incremental process, where each of the four phases contains all of the traditional SDLC workflows, such as planning, analysis, design, and implementation (which often are designated the phases in the traditional SDLC). Depending on the specific RUP phase, more or less of an emphasis is placed on any of the various workflows (George, *et al.* 2004).

In the inception phase, most of the focus is on the analysis workflow. There is some activity in design and implementation, but only a fraction of the analysis activity. In the elaboration phase, emphasis on analysis drops by perhaps 50% and design increases dramatically to take center stage. Implementation and operation are still very low in this phase but slightly higher than the inception phase. In the construction phase, emphasis on analysis drops further, as does design. Implementation becomes the main focus. In the final, transition phase, analysis and design are very low and implementation drops off slightly. In this phase, the transition from development to operation takes place. These varying levels of focus are illustrated in Figure 2.

![Figure 2: The Rational Unified Process](image)

With the basic characteristics of OOAD outlined, it is now possible to attempt the first iteration of IS Security integration into the methodology, as presented in Table 2.

After determining the appropriate placement of ISD methodological components, it is time to begin the verification process. It is here that “the devil is in the details” becomes apparent. For example, how should the IS Security issues for the actor component be integrated into the management workflow? A decision needs to be made for the actual implementation for each STD area. It may be determined that simple awareness training of responsibility, integrity, trust and ethicality for the project managers might satisfy this component. How can one verify that soft issues relating to culture, norms, and beliefs be upheld during RUP’s transition phase?
<table>
<thead>
<tr>
<th>Components</th>
<th>IS Security Issues</th>
<th>Seminal Work</th>
<th>OOAD Methodology</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structure</td>
<td>Culture Norms Beliefs</td>
<td>Backhouse &amp; Dhillon (1996)</td>
<td>Transition</td>
</tr>
<tr>
<td>Technology</td>
<td>Confidentiality Integrity Availability Non-repudiation</td>
<td>Howard (1995)</td>
<td>Construction</td>
</tr>
</tbody>
</table>

Table 2 IS Security STD framework Guiding an OOAD Security Integration

IS Security policy is a hot spot that jumps out with this first iteration of integration. With inception and elaboration being the OOAD phases where this security issue is integrated, a significant change to the typical process is introduced. Instead of fashioning a security policy in a separate or post facto fashion, it is now required to be integrated into very early phases of the ISD methodology. Requiring a synchronous security policy creation across inception and elaboration forces serious security considerations into the ISD methodology at an early and critical point.

Another hot spot that emerges is the inclusion of the techno-centric security considerations in the final phase of the methodology, construction. This follows the work of Siponen (2001) and Dhillon et al. (2000) who acknowledge the criticality of the technical aspect of security but only after behavioral issues have been addressed. Instead of worrying about encryption, access control, and password protection in the critical early phases, these issues are relegated to the end of the process. If the IS Security policy and actor issues are addressed early in the process, the technical aspects will be more likely to be effectively engineered.

This section barely scratches the surface of this first iteration of actual IS Security integration. The purpose of the paper however is simply to provide a theoretically grounded framework by which IS Security integration with an ISD methodology can be accomplished. The example provided is only meant to illustrate how one might begin the process. It is beyond the scope of this paper to provide a complete illustration.

7. CONCLUSION

This paper is intended as a starting point for future work on integrating IS Security considerations into ISD and we believe that it contributes toward a more sound
foundation. It is hoped that the framework proposed here will reduce the sometimes seemingly ad hoc nature of ISD security research.

Our proposed IS Security STD framework answers some of the questions posed in the critiques of the previous research. A theoretical foundation in the STD framework provides the rigor while the applicability of the integration method provides the relevance. Based on the review of existing literature, such a theoretically grounded and methodological approach for integrating IS Security with ISD has been lacking.

Future research may take these concepts in several directions. The most obvious area of future research would be to tackle a complete integration effort, based on the proposed theoretical framework. One could continue with the OOAD example and follow the iterative process through to a fully functional secure OOAD methodology. This new methodology (which could be called OOAD-sec) could then be tested in a case study setting. One could also take a different methodology such as SAD or SSM and perform a full IS Security integration.

REFERENCES


