Incorporating Information Assurance in Systems Analysis and Design Curricula

Stephen Barnes  
Oklahoma State University

David Biros  
Oklahoma State University

Mark Weiser  
Oklahoma State University

Nicholas C. Romano Jr.  
Oklahoma State University

Follow this and additional works at: http://aisel.aisnet.org/amcis2007

Recommended Citation
Barnes, Stephen; Biros, David; Weiser, Mark; and Romano Jr., Nicholas C., "Incorporating Information Assurance in Systems Analysis and Design Curricula" (2007). AMCIS 2007 Proceedings. 238.
http://aisel.aisnet.org/amcis2007/238
INCORPORATING INFORMATION ASSURANCE IN SYSTEMS ANALYSIS AND DESIGN CURRICULA

Stephen D. Barnes  
stephen.barnes@okstate.edu  
David P. Biros  
david.biros@okstate.edu  
Mark Weiser  
weiser@okstate.edu  
Nicholas Romano  
nicholas.romano@okstate.edu  
Department of Management Science and Information Systems  
Oklahoma State University

Abstract

The importance of teaching information assurance (IA) is now widely recognized. Universities teach the technology and tactics for detecting and preventing attacks on our systems. Hardware and software engineers are moving deliberately to increase capabilities in secure systems. Ultimately, however, these groups implement design specifications that began as an analysis of a business need or opportunity. Separate surveys of IS faculty and students reveal a recognition by both groups that IA design is critical and will be important to future employers. This paper discusses the dichotomy between existing systems analysis and design (SA&D) principles as depicted in many popular textbooks with the emerging need for secure information systems design. A review of 16 SA&D textbooks illustrates that although they provide a good foundation for the design process, they rarely include more than a page or two on IA considerations. Incorporating IA in systems analysis and design is no longer an option for our students. IS faculty must provide the formal curriculum for proper application of IA technologies in final systems implementation. We propose embedding IA in all sections of SA&D curricula, and provide a description of our approach to testing a methodology we are developing for secure SA&D courses.

Introduction

The problem of digital insecurity is pervasive and widespread. Much of the reason for this is that many systems were never designed with security in mind. When systems are developed, project teams are focused on cost, deadlines, and stakeholder concerns. Rarely are information security concerns considered in the development process. This is evidenced by the 15,971 Common Vulnerabilities found at the United States Computer Emergency Response Team (US-CERT) National Vulnerability Database (US-CERT 2006). Software flaws are so commonplace that Microsoft, the world’s largest software company, schedules regular monthly vulnerability announcements about its products flaws (Microsoft Corp. 2006). Security design flaws are both expected and, unfortunately, accepted.

In recent years there has been a greater emphasis placed on information assurance (IA, used synonymously with information security here) in education. The U.S. Government further stimulates IA education through its Center of Academic Excellence in IA Education (CAE) program.
These recognized IA programs typically grow out of the curriculum of Computer Engineering, Computer Science, or MIS. Traditionally, these are engineering topics focused on the detection of security breaches and protecting weak systems from being reached or accessed with devices such as firewalls and proxies. Recently, an emphasis has been placed on secure coding and compiling to address fundamental and widespread issues such as buffer overflows.

These are indeed important issues, however, IA principles are weak or non-existent during the systems analysis and design (SA&D) process that precedes coding, and is the traditional domain of MIS education programs. It is during this process that security issues must first be addressed in order to avoid most IA problems from happening, rather than addressing them after the exploit. Unfortunately, secure design is neither a part of the Federal certifications nor is it a significant part of any of the SA&D textbooks. This creates a “garbage in / garbage out” scenario – weak or non-existent IA incorporation in SA&D means that even excellent hardware and software implementations exactly meeting design specifications are likely to be flawed.

This paper explains the importance of including IA techniques in the SA&D process. Through the results of a survey of faculty and students from different universities, as well as reviews of current courses and SA&D textbooks, we demonstrate the need for IA in the design process. This paper proposes a fundamental change to SA&D that embeds IA in every phase of the process. We describe how we are testing a methodology that can be used in the classroom and practice to integrate these changes into the long-standing Systems Development Lifecycle (SDLC) model and incorporate these changes into standard textbooks on the topic.

**Systems Analysis and Design and Information Assurance**

The developments of systems analysis and design methods and IA methods have progressed along separate lines. SA&D methodologies have evolved during the last three decades with a focus on structured methods and rapid development to meet organization’s needs and desires. IA is a more recent development and has often focused on perimeter security of the network and identification of intruders. Unfortunately, the design ideals often differ between systems analysts and IA specialists (White and Dhillon 2005). While security is beginning to be referenced in SA&D textbooks, it has yet to be integrated into the formal process, in spite of a general recognition that the process is appropriate (White and Dhillon 2005).

**Evolution of the SA&D Process**

Early digital system development processes lacked formal methodologies and didn’t produce optimal results. As requirements complexity increased, modern structured analysis and design methods (i.e. SDLC) developed. Eventually, modern structured analysis and design methods such as those developed by DeMarco (1979), Yourdon (1989) and Chen (1976) entered the scene. Later, object-oriented methods were developed (Booch 1994; Coad and Yourdon 1991a; Coad and Yourdon 1991b; Embry et al. 1992; Rumbaugh et al. 1991), only to be followed by faster methods such as rapid prototyping, rapid application development (RAD) and joint application development (JAD) that provide more expedient models of development. Nonetheless, the basics steps of the waterfall model are still considered in these newer approaches and provide a good foundation of the SA&D process, and the SDLC still serves as a useful baseline from which to teach new students a structured approach to SA&D.

**Evolution of Information Assurance**

In the past 10-15 years, IA has grown in prominence in information systems use. In a recent Gartner EXP survey of over 1300 CIOs in both the public and private sector, security was ranked second as one of the top 10 business priorities for 2005, and data protection and privacy placed fifth (Government Technology 2005). These organizations spend millions of dollars each year protecting their information systems and administering patches to security vulnerabilities.

Universities and training organizations have responded to market demands with a number of IA programs that are mapped under the Information Assurance Courseware Evaluation (IACE) program. The majority of these certification areas are methods to detect breaches and provide technical mechanisms to prevent access to system weaknesses, such as buffer overflows, but none provide any structure for security throughout the analysis and design process.
Dichotomy Between Systems Analysis and Design and Information Assurance

Organizations are increasingly aware that their information systems are subject to many threats because the systems were not designed with security in mind. There is a realization and acceptance that more secure code is required, but that must start with better design. Unfortunately, the goals of development managers and systems analysts often differ than those of IA professionals (White and Dhillon 2005). Designing a system with security in mind can slow down the process. On the other hand, systems with numerous vulnerabilities require patches or fixes and time to mitigate them, driving up costs and causing operational delays. Unless security processes are integrated into the design process, this dichotomy will continue to exist.

SA&D Curriculum Materials

This paper holds that the addition of IA education in SA&D classes may serve as a partial answer. Training and education are often driven by the content of relevant textbooks. Unfortunately, the most widely used SA&D textbooks only touch on information security, rather than fully incorporate it as a critical part of the process. One book goes so far as to state that

“...the (security) issues arising from the development of web-based information systems are extremely important. Some of them are remarkably hard to solve. But none of them impact the task of the systems analyst.” (Schach 2004)

We reviewed 16 of the current classical approach (Dennis et al. 2006; Harris 2003; Hoffer et al. 2002; Kendall and Kendall 2004; Marakas 2006; Satzinger et al. 2004b; Shelly et al. 2006; Valacich et al. 2006; Whitten and Bentley 2007) and recent object-oriented SA&D texts (Dennis et al. 2005; Lau 2000; O'Docherty 2005; Schach 2004) for their security and IA content1. Only one (Satzinger et al. 2004b) contained a full chapter on “Interfaces, controls, and security”, while the remaining texts had anything from one sentence to six pages of relevant material. None exceeded 3.2% pages with IA coverage, with most under 0.5%, and none appeared to integrate these topics into the pertinent sections of the text (e.g. seeking out access control policies as a part of the requirements gathering process). In addition, we reviewed three older texts that were relevant to this analysis (Ostle and Arnold 1985; Senn 1984). One (Ostle and Arnold 1985) contained an excellent chapter on security and protection measures. Table 1 summarizes our findings for six popular SA&D texts on the market today2.

<table>
<thead>
<tr>
<th>Classical Texts</th>
<th>%3</th>
<th>Information Assurance Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dennis, Wixom et al., <em>Systems Analysis Design</em>. 3rd ed., 2006. (2006)</td>
<td>0.54%</td>
<td>Includes a three-page section in the architecture design chapter on security requirements. Topics in this section include: availability requirements; system value; access control requirements; encryption and authentication requirements; and virus control issues.</td>
</tr>
<tr>
<td>Hoffer, George et al., <em>Modern Systems Analysis and Design</em>. 3rd ed., 2002. (2002)</td>
<td>0.41%</td>
<td>Includes one page on database integrity and controls, focused on how to repair damaged data with backups, audit logs, and recovery processes. Security is mentioned, suggesting that encryption, authentication, authorization, and manipulation controls provide the basics needed.</td>
</tr>
</tbody>
</table>

---

1 We are also aware of the following texts: Coad and Yourdon (1991a); Coad and Yourdon (1991b); Blaha and Rumbaugh (2004); Satzinger, Jackson et al. (2004a); Stumpf and Teague (2004); Arlow and Neustadt (2005); and O'Docherty (2005). We were unable to obtain copies of these volumes, and thus were not able to evaluate them for this study.

2 Please contact the authors for a more comprehensive table.

3 Column shows ratio of pages including some IA content to total pages, excluding front matter.
Without a fundamental change to our SA&D models, core curriculum, and references, the mindset of the next generation of designers will continue to lack a security focus, requiring significant and expensive post-implementation security approaches.

**Perceived Need for IA in the SA&D Classroom**

We surveyed two groups to learn their attitude towards IA in the SA&D classroom. The first group consisted of MIS faculty and IT professionals gathered at a pair of IA-related research conferences. The second group consisted of an undergraduate SA&D class at a large state university in the Midwest. The results of this student survey are discussed after we introduce our proposed changes below.

**Faculty/Professional Response**

Twenty-eight faculty and IT professionals with IA knowledge from 20 different organizations were surveyed about their perception of IA’s place in analysis and design, its importance on the job market, and the means through which it should be and is presented to SA&D students. Every respondent felt that an IA education was “very important” or “important” for someone pursuing jobs in systems analysis and design, with 82% of respondents prioritizing it among the upper half of all the systems analyst’s tasks, and perceived that business employers are either “very concerned” or “moderately concerned” about IA.

Faculty, as part of the surveyed group, perceive IA as critical to a proper SA&D education and future employment opportunities for their students. Seventy-five percent of these same faculty felt that the proper way to address information assurance was to integrate it into each section of the development lifecycle, although only 5% reported that as their school’s approach. 43% indicated that IA was not taught within the SA&D course, but was available in a separate class. In fact, only two surveyed schools indicated a significant amount of IA in their SA&D course (greater than 20% of the content). Both of those respondents, however, estimated that 75% of IA material came from sources other than the primary text, a model that is extremely unlikely to be generalized to all universities and training organizations. The extra effort by a minority of schools to address IA issues in SA&D is admirable, but unlikely to be adopted in a large number of schools until formal methods exist and are documented.

**Undergraduate Response**

Students with awareness of both IA and SA&D recognized the value of embedding IA in SA&D education. A similar survey was given to SA&D students at a large state university in the Midwest. Regardless of the amount of prior IA training, students felt they understood more about how to apply IA in SA&D after our IA introduction, with 39% believing they learned “a lot more” under this structure. This learning was also valued – 95% perceived covering IA in the SA&D class to be either “valuable” or “very valuable” and most felt that it should be fully integrated in each section of the development lifecycle.

Because most students at this level are actively seeking employment, we investigated perceptions of these future jobs. Every student recognized that future employers would be “very concerned” about information security and ranked IA in the upper half of systems analyst priorities. Every student recognized that future employers would be “very concerned” about
information security and ranked IA in the upper half of systems analyst priorities. Both faculty and students recognize the importance of IA and SA&D and the lack of it in most curricula. A fundamental shift must be made to recognize and capture security constraints early in the design process, breaking or limiting the “build and patch” cycle.

Testing the Approach in the Classroom

The importance of performing formal logical modeling lies in the inherent validation of system requirements that it provides. During this part of the development process, the analyst deconstructs the requirements of the system to identify all the data entities and their relationships and ensures that it is possible to create the required outputs given the provided inputs. Anomalies are almost always discovered during modeling and can be much more easily addressed well before the first line of code has been written. To test this, we are currently testing a comprehensive set of mechanisms that unobtrusively integrate information assurance requirements in the design process by integrating it into SA&D courses at our institution.

A Secure Modeling Approach

The specifics of our proposed modeling approach will be detailed in a separate paper. Briefly, the approach extends each of the main steps of the SDLC process to include aspects of the security model. These include:

- Extension of the requirements gathering process to seek out the relevant security constraints,
- Techniques for capturing the sensitivity of data and processes on the data flow diagrams and the entity relationship diagrams,
- Integration of security constraints in the data dictionary as it affects user access to specific entities, attributes, and processes,
- Some discussion of how to utilize secure channels at the physical design stage,
- Integration of security into modern graphical user interfaces, in particular the currently available browsers,
- Use of authentication and authorization tools and techniques,
- And a discussion of the impacts of these technologies on the time and costs of developing a secure application.

In addition, a series of examples are developed around how to integrate these lessons into the project. For example, embedding IA in data modeling adds a few rules to the typical process, including:

- No sensitive attributes will be used as primary keys
- Use substitute candidate keys where necessary
- Where possible, eliminate or encrypt sensitive attributes
- Propose tight typing and review with client
- Propose risk mitigation methods for sensitive attributes.

The modeling approach we are developing includes modifications or extensions to a variety of commonly used notations, including use cases for requirements gathering, ERDs, DFDs, and the data dictionary, and involves marking any sensitive process, data element, or system with notations that make it clear to the downstream coding team that special handling is necessary. Every item that is noted with “sensitive” flag must be addressed in some fashion, with a suggested way to mitigate the risk presented by that item. The proposed intervention is dependent upon the requirements for the use of the individual attribute. Some authors put this process in a later design phase. We not only protect sensitive data and processes with this approach, but we expect that it will also limit many application-layer exploits.

An example may be helpful. For instance, the SSN must be occasionally used to locate a student and their respective enrollments for the purposes of financial aid; however, there is no explicitly stated reason why any user would need to generate student SSNs as an output. In this example, the analyst may recommended that a one-way hash of the input be stored, rather than the SSN itself. The short and fixed length of an SSN is clearly vulnerable to brute force attack. This one-way encryption, however, along with the use of a substitute candidate key to limit the appearance of the Social Security Numbers is judged by the analyst as sufficient mitigation to lower the risk of this element. When revisiting the analysis with the clients, a discussion of the residual risk would be held to see if they would like to accept the remaining risk, impose further technical restrictions, transfer the risk in some manner, or decide to avoid the risk by finding a way to eliminate storage of SSNs. Such a tight data specification process makes later design and implementation processes easier, quicker, and less prone to inadvertent addition of vulnerabilities.
The Classroom Quasi-Experiment

Traditionally, SA&D is taught with an accompanying project that applies the principals learned in the lectures to real problems. We have accomplished this with a variety of methods, but have recently been using written case studies at our institution. This bounds the problem, and ensures that each team of students faces similar problems.

To test the proposed modeling approach we are teaching the security modeling process to one section of our SA&D course as a fully integrated process, while providing the same basic materials minus the security information in two other (smaller) control sections. The control sections will spend a week at the end of the term catching up on the security information, taught as a separate module. These represent the “integrated” approach, and the “bolted on” approach, respectively.

For this quasi-experiment, we have tweaked our case study very modestly, to incorporate reasons that some data and process should be kept secure, and will ask that students include the relevant elements of the security methodology in each of the main project units. The control group gets the same case study, but the assignments do not include the security specific questions. All the remaining lecture and supporting materials are the same for each class, including the use of a common textbook. Due however to administrative constraints, the treatment section instructor is not teaching the control sections.

A pre-test will be given to every student, included in a survey of IS and IA classroom and work experience. This will allow for control of later evaluations of project and homework based on prior knowledge of the relevant topics. The main experimental sampling will consist of an evaluation of each of the team projects for their security implementations and the adequateness of their designs in meeting a set of client goals that were set out in the case materials provided as the project proceeded. A separate but similar evaluation based on IA questions included on the final exam may also be possible.

Finally, we plan to survey the students for their understanding of the materials in a non-graded and voluntary manner, specifically to get their feedback on the approach and understanding of the materials. This will include specific questions asking for their suggestions on how to improve the approach, and where they saw weaknesses in it or the related materials presented. This leads, of course, to a qualitative analysis of the results of our work that can be coupled with the more objective evaluation of the resulting project work.

Discussion

This paper posits that good system information security begins with design. Patching systems to address information security flaws is simply inefficient and expensive. We applaud efforts to create more secure code, but without system requirements and business rules for IA, problems will remain.

Managers, students, and MIS faculty alike have noted a need to increase IA consideration in the SA&D process. As we have shown, many of the popular SA&D textbooks provide some discussion of the necessity to include IA in system development. The current analysis and design environment has project managers struggling between enhancing systems security and delivering a desired system on time and within costs. Our approach is simple and powerful, embedding IA in the process, rather either omitting it, or requiring complex additional steps. Although incomplete as discussed here, a full set of SA&D security modifications will help break the “build and patch” cycle. Overall implementation costs will be driven down through tighter specifications to programmers. Further, lifecycle maintenance and security costs will also be lowered because many of the vulnerabilities will have been addressed in analysis and design and securely coded in the final implementation.

For these reasons and others, MIS faculty must weave IA into SA&D in order for it to be widely adopted as simply part of the process. In order for that to be widely adopted, these mechanisms must be embedded in the next generation of textbooks, or made available in companion books, with information that can be applied in every area presented in the main textbook. Only then will graduates be available that can help industry truly leverage efforts in teaching technical security methods and new initiatives in securing the programming function itself.

References


Satzinger, J.W., Jackson, R.B., and Burd, S.D. *Object-Oriented Analysis and Design with the Unified Process* Course Technology, Boston MA, 2004a, p. 656.


