

Web Engineering Security: A Practitioner's Perspective

William Bradley Glisson
Department of Computing Science,
The University of Glasgow
Glasgow, G12 8QQ, Scotland
+44 (0)141 330 4256 (ext. 0995)
glisson@dcs.gla.ac.uk

Andrew McDonald
Department of Computing Science,
The University of Glasgow
Glasgow, G12 8QQ, Scotland
+44 (0)141 330 4256 (ext. 0995)
andrew@dcs.gla.ac.uk

Ray Welland
Department of Computing Science,
The University of Glasgow
Glasgow, G12 8QQ, Scotland
+44 (0)141 330 4256 (ext. 4968)
ray@dcs.gla.ac.uk

ABSTRACT

There are a number of critical factors driving security in Web Engineering. These include: economic issues, people issues, and legislative issues. This paper presents the argument that a Security Improvement Approach (SIA), which can be applied to different Web engineering development processes, is essential to successfully addressing Web application security. In this paper, the criteria that any SIA will have to address, for a Web engineering process, are presented. The criteria are derived with supporting empirical evidence based on an in-depth security survey conducted within a Fortune 500 financial service sector organization and supporting literature. The contribution of this paper is two fold. The criteria presented in this paper can be used to assess the security of an existing Web engineering process and also to guide Security Improvement Initiatives in Web Engineering.

Categories and Subject Descriptors

D.2.9 [Software Engineering]: Management – *software process models*; K.4.4 [Computers and Society]: Electronic Commerce – *security*; K.6.5 [Management of Computerised Information Systems]: Security and Protection.

General Terms

Design, Security

Keywords

Web Engineering, Software Engineering, Security, Survey, Development Process

1. INTRODUCTION

The results of industrial surveys, over the past couple of years, indicate that economic losses resulting from security issues are staggering. The 2004 CSI/FBI Computer Crime and Security Survey estimates that losses from internet security breaches, in the US, exceeded \$141 million while the 2005 CSI/FBI Computer Crime and Security Survey estimates total losses exceeded \$130 million[8, 11, 12].

Although the results of the FBI survey indicate that the total cost has dropped from the previous year, there are still security issues that are costing organizations significant amounts of money. The 2005 FBI report specifically identifies unauthorized access and the theft of proprietary information as sources of considerable financial losses[12]. In concert with the 2005 FBI survey, the 2005 Deloitte Survey indicates the security cost is still

significant[3]. The report indicates that identity theft cost the UK almost a billion dollars in 2003[3]. To complicate matters, the survey indicated that threats are becoming more sophisticated and that there is a severe lack of employee awareness[3]. The report even states that “it is clear that many security breaches are the result of human error or negligence resulting from weak operational practices”[3]. Over the past several years, additional pressure is being applied to the business community from a legislative perspective demanding secure web applications[9]. Even with the increased legislative pressure, the accuracy of the industry surveys is questionable due to the fact that most organizations do not want this information made public[9].

In addition to the published tangible losses from security breaches[3, 8, 11, 12], there are significant intangible losses and associated business impact to organizations, where inadequate or perceived inadequate approaches to security have materialized. These intangible losses can take a number of forms, such as: delays to release dates for products; retraining of staff; reduced consumer confidence in a supplier and their products; and the need to re-engineer processes to enhance and improve security. For example, during the first quarter of 2002, Microsoft announced a major initiative to address security throughout the company as a result of a number of security incidents highlighted previously in the media [16]. It is not known exactly what impact this initiative had or continues to have upon Microsoft's bottom line. However, the following quote from Microsoft's director of security assurance at that time indicates the wide spread impact of the initiative upon Microsoft's operating approach “We are going to get a lot of testing done. We are going to have a lot of people who are really, really hard-core about security distributed throughout the organization, and that's going to change how products get built in the future.”[16]

The market is producing economic support for an idea, as quoted by Steven R. Rakitin, that W. Edwards Deming put forth several years ago stating that “The quality of a product is directly related to the quality of the process used to create it”[23]. One of the major differences between Web application development and conventional software development is a greater emphasis on security[4]. Hence, the increase in costs associated with security issues should raise concerns over the way security is addressed in the Web application development process.

This information presents economic issues and legal issues that provide valuable evidence supporting the need for the development of a Security Improvement Approach (SIA).

In order to acquire a more in-depth understanding of the challenges and the issues that face a large organization during Web application development, a survey was conducted in a

Fortune 500 financial service sector organization. The name of the organization is being withheld to ensure organizational anonymity. Following this line of thought, the names of the documents, the names of the processes and the names of the groups have all been altered. The results of the interviews with employees are presented anonymously. Maintaining organizational anonymity helps attain accurate information and creates an environment where all parties are comfortable presenting commercially sensitive data.

This paper summarizes the results of the Fortune 500 security survey establishing the justification for a Security Improvement Approach (SIA) and the essential criteria while specifically focusing on how this relates to security within the Web engineering process.

2. SURVEY ANALYSIS

The survey examines security from the overall application development perspective and from the perspective of security within the process. This survey was conducted in the same organization as the research for the Agile Web Engineering (AWE) process and the new results from the application development piece of the survey support previous findings[17]. In-depth survey details are available in our recent technical report[10]. The following sub-section describes the rationale behind our experimental approach.

2.1 Methodology

Zelkowitz and Wallace [26] describe and present a taxonomy for software engineering experimentation that comprises twelve different experimental approaches. Each of the twelve experimental approaches are categorized into one of three broad categories: observational methods collect data as a project develops; historical methods collect data from projects that have already been completed; controlled methods provide for multiple instances of an observation for statistical validity of the results. Due to time and cost constraints, the Lessons Learned approach from the historical category, that “examines qualitative data from completed projects”, [26] was adopted. This took the form of a series of structured interviews using a qualitative one-to-one interview technique for gathering the opinions and experience of others during Web application development. This approach has the advantages of enabling the determination of trends and is inexpensive[26]. However, it does not allow for the production of quantitative data and constraining factors[26]. A historical approach was selected to help the authors understand how security challenges and issues had been perceived during recent projects within the company.

2.1.1 Interviewee Demographics

Within the organization, sixteen interviews were conducted. This survey sample consisted of various employees representing a variety of roles with a diversity of work experiences within the technical side of the organization. The initial questions were used to establish the interviewee’s current role in the organization; his/her number of years of experience and a brief idea of the individual’s history. These questions revealed that the interviewees are experienced IT professionals who have a variety of technology backgrounds; and, in general, several years of experience. The average number of years of experience among the sixteen respondents was just under fourteen years. To comprehend

the security challenges the application development process was examined first in order to understand the environment. Then the security implications of the environment were scrutinized.

2.2 Application Development Process

The web application development findings that are of particular interest to web engineering security research are as follows:

- The overall organization uses a customized plan driven document centric waterfall approach for all application development including Web applications.
- After going through a formal design approval process there is no verification that the design implemented in production is the design that was originally approved.
- It is questionable as to whether the development process is always followed.
- Realistically, the organization is operating two different approaches to application development at different levels within the organization. The high level approach is a customized version of the plan driven waterfall approach. The low level approach is a number of ad-hoc processes contrived by the individual coding teams.
- The current application development process is not effective when considering time-to-market issues, rapid application development needs and the introduction of new technology resulting in a lack of efficiency.
- The general indication from the interviewee answers is that projects exceed estimated budgets and time frames on a fairly regular basis.

2.2.1 Security within the Process

Interviewees were asked about where security is involved in the development process. The results of that inquiry are summarized in Table 1. This revealed a severe lack of consideration for security in the business analysis stage of the development life cycle. It also indicates that there are deficiencies in the Evaluation, Deployment, and Maintenance and Evolution stages.

Table 1. Security Involvement Table

STAGE	YES	NO	OTHER
Business Analysis	4	9	3
Requirements	10	1	5
Design	13		3
Implementation	9	4	3
Testing	9	3	4
Evaluation	5	5	6
Deployment	9	4	3
Maintenance and Evolution	6	5	5

* Other is any answer that was not a YES or a NO

The variety of answers that were received when asking employees where security was involved and identifying the sponsor who was responsible demonstrates the lack of practical security knowledge within the organization. When asked specifically about a security process, the majority of the respondents indicated that there was no documented process. However, when asked if someone was responsible for security within the organization, six out of the eleven positive respondents named a variation of the risk team.

This is a firm indicator that security is viewed as someone else's problem within the organization.

Actually, the company does have a documented security process in the Project Risk team. The interviewee responses reveal that the knowledge of the document is restricted to specific groups. Of the five "yes" answers to the existence of a security development process, it was unanimous among those five respondents that the security development process applies to all types of application development, including Web development projects.

The problems that were discussed concerning the current security process included a lack of emphasis on the employee, a lack of utilization of that process, a lack of security involvement after the design has been signed off, and a lack of security awareness and stakeholder buy-in to security. The point of break down appears to be the length of time around the entire development process. The business has the power to circumvent the process to keep projects on track from a time line and budget perspective.

One of the thoughts behind the lack of a known security process within the organization seems to be around the fact that the individuals involved in security do not record the process. They just do what needs to be done. These people are viewed as a resource and are accessed as needed during the development process. However, there is some confusion over when and where the Project Risk team actually gets involved in the process. This is taken to the point that security is viewed as the architect's problem. There is also the view that security is a bolt-on issue that is addressed after the coding is complete. Hence, the organization is giving security lip service only and is not truly pursuing a security architecture infrastructure.

2.2.2 Security Determination

When asked how applications are deemed secure within the organization, the answers ranged from requirements, to policies, to security standards, to processes, to testing, audits and reviews. Requirements refer to the business and the application requirements. The policies and standards are set by the Project Risk team and industry standards are used to help ensure security within the organization. The process refers to the creation of the Design Architecture Document (DAD) and submitting it to the Design Architecture Committee (DAC). The testing from the organizational perspective refers to internal penetration testing and third party testing. Testing from the development perspective is subjective and tailored around the needs of the application based on the functional and non-functional requirements. The general rule is that high risk applications require more testing and, potentially, third party testing.

The answers indicate that the test used on specific applications depends on the needs of the individual application. Outwardly facing applications (i.e. Web Applications) are more rigorously tested than inwardly facing applications. Some issues related to in-house testing did surface through conversation generated via the survey. Some of the respondents indicated that time losses occurred between testing windows. If the start time for a specific test is missed, the respondents indicated that it could be as long as two weeks before another testing opportunity could be seized.

When it comes to testing, audits and reviews, as far as the criteria applying to all applications, the general consensus was that it depends on the environment, the amount of risk presented and the

application facing that determines the security criteria that would be applied.

The survey confirmed that conflicts arise between the stakeholders and the individuals responsible for security. Fourteen of the respondents indicated that conflicts arise between the two groups. The types of conflicts range from financial and time constraints, to conflicts over security solutions. The disagreement over the security solution appears to have its roots in the perception of the level of risk that is perceived with an application. Hence, a higher level of risk would necessitate a stronger security solution. This disagreement about perceived risk could logically take place between the business unit and the application developers. An interesting point that did surface is that certain business units also have their own individuals specifically assigned to evaluate the risk a new application presented. When there are conflicts on the analysis of a project's potential risk, this work environment has the potential to exaggerate disagreements between the technology area and the business unit.

The survey revealed that contractors are used heavily in the organization. The majority of the respondents indicated that contractors are held to the same application development methodology as employees. If they use a different process, then the process is examined and approved by the organization. The majority of the respondents indicated that contractors are also held to the same security requirements as employees. However, reading between the lines in conversation, the organization does not consistently test contractor constructed applications. Hence, there is the possibility that there are discrepancies in application testing. How effectively this is monitored and addressed appears to be up to the discretion of the project manager.

2.2.3 Practical Security

When the interviewees were asked their opinions on the emphasis security plays within the organization, some individuals think that the emphasis on security is strong, due to outside factors such as legislation, while others feel that the emphasis is weak. A couple of individuals feel that the emphasis has improved over recent months. While others feel that the security focus is still misaligned. Some individuals feel that security plays a large role in the organization while others feel that the emphasis is small and that security is effectively seen as an inhibitor rather than an enabler in the development process.

Drilling down to the heart of the matter, an attempt was made to determine if the elements of the existing in-house security process are always followed. The result is that seven out of the sixteen respondents indicated that it was not always followed. There was one "sometimes" answer and the rest indicated that it was always followed. The interesting point is that there were only five respondents indicating that a process exists but there were eight individual solid "yes" answers and one "sometimes" answer that indicated that the elements of the in-house security development process were always followed. This indicates that there is at least an implicit security development process, or interviewees felt that it was politically correct to say that it is always followed even if it is perceived not to exist or is not understood! The reasons for not following the development process range from time pressures, to bureaucracy, to lack of awareness, to a lack of security involvement in certain aspects of the process. Other reasons that were mentioned include a complete lack of a process and where

the application sits, i.e., does the application face the internet or is it internal.

Eleven of the individuals who were surveyed feel that security should play a larger role in the organization's development environment. Four of the individuals surveyed feel that the current role security plays in the development environment is accurate and one feels that there are cases where it should play a smaller role. The individuals who feel that the role should be larger base their opinion on several different reasons. The reasons that seem to recur throughout the answers to this question are focused around the business. They indicate that the financial world is a relatively small world and protection of the reputation is critical. In the current environment, security can be de-scoped due to numerous reasons; integrating security into the development process up front would cut development overhead and increase security awareness within the organization. Various views on the accuracy of the current security role included a good balance between security and the development environment; that the current role meets project needs; a need to extend security throughout the development life cycle and a need to engage the Risk Team as early as possible.

Eight out of the sixteen individuals surveyed feel that there is no job related impact for not following the development security process. Two of the respondents indicated that they do not know if there is an impact and six of the respondents feel that there is a job related impact.

2.2.4 *Perceived Threats*

An attempt was made to determine major threats to the organization during application development. There were a variety of answers that ranged from "ignorance, naivety, and incompetence of the people implementing the technical solutions", to coding issues, to coder issues, to general management issues.

One of the respondents obviously questioned the skill level of the individuals who were creating and implementing the design and the security model of the proposed solution. This was echoed via other interviewee responses but with more political correctness. The coding issues that were discussed seemed to focus on the production of bad code. This could be caused by completing code rapidly, bad coding practices, not understanding requirements, or malicious activity on behalf of a developer.

The issues around the coder seemed to focus on the dangers associated with contractor reliance. Reliance on an outside contractor creates vulnerability from a coding practice perspective and from a skill set perspective. If you do not have the skills in-house to support the product and the contractor leaves, then the organization has to scramble to replace that individual at the risk of a high cost. This also brings up another issue that surfaced in this line of questioning, and that is, single developer reliance and high contractor reliance. This indicates that the organization does not do a good job of sharing development knowledge.

The managerial issues seemed to focus on unreasonable time scales and poor project management from a time and budget perspective. The unreasonable time scales imply a lack of understanding of the project requirements on the part of the manager. The poor project management of time scales and

budgets is inevitably going to put pressure on the coding teams to produce a product within shortened time scales.

Security must address the people issue, indicating that there has to be a way to establish trust with individual employees and maintain trust with those employees. The process needs to be examined from an end-to-end-perspective to be sure that it delivers the desired results. These results need to be examined from a product, a security and both an effectiveness and efficiency perspective. The results of the survey support the idea that there are fundamental security problems with the methodologies being used in real world web application development.

Education is an important area of the security process. Security education should not only include raising awareness of the different types of technical attacks and social engineering attacks [8, 21], but it should also include information about the current environment. Employees should know with whom they should discuss security, how it fits into their everyday work environment (i.e. their development process), and the potential impact security has on the Web application solution that they are proposing / introducing into the organization.

When the interviewees were asked about the issues they thought were being met and the ones they thought were not being met, a variety of answers were received. The answers ranged from the coding issues being addressed within the company, to a good implementation of separation of duty, to a lack of completely re-testing applications when updates are implemented, to out-of-synch testing and production environments, to a lack of specific security skills.

When asked about areas that require more or less emphasis, some of the recurring themes included business requirements, education, and testing. When the interviewees were asked about the major security risk that they perceived during application development, the range of answers included these common themes; seven mentioned code/design/testing /requirements, three mentioned people and behavior, two mentioned policy circumvention and enforcement, and two mentioned viruses. There were a variety of answers to the question inquiring which of these issues are being met by the existing process, which ranged from none to all. A few individuals did indicate that separation of duty, code reviews and testing is sufficient within the organization. There were several respondents who indicated that issues were not being satisfied by the existing process. Other answers ranged from a lack of documentation, to internal and external coding issues, to a lack of security in the design architecture.

The survey confirmed that conflicts arise between application developers and the individuals who are responsible for security. Hence, security from time-to-time is perceived as the culprit when Web application development projects do not hit pre-determined goals. This supports the thought process behind implementing security from the beginning of the project and sustaining it throughout the life of the project. Integration of security early in the development process helps move security from a perceived application development blocker to that of an application development enabler role.

3. RELEVANT WORK

Industry surveys recognize the importance of security in reference to the World Wide Web[2, 3, 12]. The industry surveys provide great information on the amount of money that is being spent by the industry along with insight as to the high-level issues that affect the security industry. However, they do not drill down into individual organizations in order to uncover specific security issues within the development process. Nor do they recommend possible solutions to the issues that are uncovered through detailed development process security surveys.

Industry recognition has prompted security improvement articles and white papers. For example, Taylor and McGraw [25] propose an improvement program that offers very good, very broad business advice which does not go down to the detail of the development process. In a paper referenced in the previous mentioned article McGraw makes valid points in that “we must figure out ways to build easier-to-defend code” and the real problem “is poorly designed and implemented software”[19]. The article indicates that the practices listed are “process agnostic”[19], however, this is not readily apparent from a web engineering application development perspective. Additional issues include the fact that the security discussion starts at the requirements stage, not the business analysis stage indicating that there are issues with security visibility, no discussion of organizational buy-in, delivery, or trust and accountability.

Systems Security Engineering - Capability Maturity Model (SSE-CCM) presents a document intensive best practices highly structured model solution designed to support statistical process control to all forms of software engineering[24]. The SSE-CCM version of the life cycle includes concept, development, production, utilization, support and retirement stages[24]. This all-inclusive approach is composed of twenty two processes. The first eleven process areas focus on security and the last ten focus on “project and organizational activities”[24]. The process areas that focus on security provide a high level initiative telling organizations what to address, for examples Coordinate Security. Under Coordinate Security they indicate that “all members of the project team are aware of and involved with security engineering activities to the extent necessary to perform their functions”[24]. This statement focuses on the team not the methodology that is being used. While there is minimal concept commonality, in general, the scope of SSE-CMM is much broader than our work.

The following section sets out the security criteria for Web Application Development. These criteria are supported directly with evidence from the survey and supporting literature.

4. SECURITY CRITERIA FOR WEB APPLICATION DEVELOPMENT (SCWAD)

Industry surveys have established the global problem and the organizational survey has established the local problem in developing secure Web applications. Together, they support the need to establish a Security Improvement Approach (SIA) that can be applied to different Web engineering development processes. In order to accomplish this goal, a set of criteria needs to be established that is specific to Web engineering processes.

The analysis of the information generated in this paper is reported in the Web Engineering Security (WES) Application Survey Technical Report[10]. Exler makes an excellent point in “*Security*

and the Application Development Process” in that “the best protection” during application development “comes from a bulletproof, practical, rigorous, and scalable process that includes security”[6]. The questions then becomes what does an organization use as a guide to achieve the best protection and how does an organization effectively critique a Web application development process? The answers to these questions are derived from the Fortune 500 financial survey discussed in the previous section and relevant literature. SCWAD identifies six essential security criteria within methodologies:

1. Active organizational support for security in the Web development process
2. Proper Controls in the development environment
3. Security visibility throughout all areas of the development process
4. Delivery of a cohesive system, integrating business requirements, software and security
5. Prompt, rigorous testing and evaluation
6. Trust and Accountability

4.1 Active Organizational Support

Active organizational support for security in the Web development process is critical. Without the support of management, there is no hope for effective integration of security within the development process. Managerial support for security needs to be both proactive and reactive. Management needs to be proactive by supporting employees, hence, giving them the necessary tools to be successful in their endeavors. Likewise, management needs to be reactive by stating and enforcing job repercussions if employees do not follow security practices within the development process or the development process in general upon which the security process depends. This lack of enforcement is blatant in the organizational survey through the number of respondents who indicated that there was not a job related impact associated with not following the development process. This also means that the development process and any existing security measures apply to all employees including contractors and permanent employees. Again this issue is questionable in the organizational survey.

Active organizational support includes encouragement of security communication among employees. The process itself should encourage communication among employees. The increased communication should translate into a better working understanding of the role security plays in the development process and the organization. This better understanding should increase the overall level of security in the development process. A key component of security is education. Employees need to be formally educated on the role security plays in the development process and in the organization, i.e., all stakeholders need to understand all of the security requirements along with the role security plays in the development process.

4.2 Proper Controls

The organization has to have proper controls. The term proper controls is a very broad term that encompasses policies, knowledge, technology, and processes. These controls help to provide structure to the development environment.

4.2.1 Policies/Standards/Procedures

Policies, standards and procedures are utilized to assist in providing a cohesive organizational infrastructure. “The goal of

an information security policy is to maintain the integrity, confidentiality and availability of information resources"[14]. The policy indicates "what" is to be done while the standards and procedures indicate "how" it is to be accomplished[13]. The detail to which these controls are developed and implemented will depend on cultural and business DNA of an individual organization. The organizational survey revealed that development and security policies have been created and are maintained in the company. However, if the development process is not always adhered to, then it stands to reason that the policies are not always enforced. If a project goes through under the wire, there is no guarantee that the security team has been properly briefed on the project details.

4.2.2 Knowledge

Organizations need to encourage knowledge transfer among employees and provide for proper training. Such training is necessary with respect to comments regarding incompetence covered in the *Perceived Threat* section of the paper, issues with coding, issues with the coders themselves, and managerial issues, all of which impact security issues in Web Development.

The Organization for Internet Safety (OIS) publishes Guidelines for Security Vulnerabilities Reporting and Response in which they define a security vulnerability as "a flaw within a software system that can cause it to work contrary to its documented design and could be exploited to cause the system to violate its documented security policy"[8, 22]. This translates into the fact that any flaws in the system design or application coding can potentially lead to security vulnerabilities[8]. This problem is emphasized due to the availability and accessibility of Web applications. Common Web development security problems include un-validated parameters, cross-site scripting, buffer overflows, command injection flaws, error-handling problems, insecure use of cryptography, and broken access controls [1, 8, 20]. Hence, designers and developers should be educated on common development flaws, best coding practices and the implementation of practical development solutions. Coupling the OIS definition with the results of the survey supports the idea that security can not be left to the acquisition of the functional and non-functional security requirements. It also supports the idea that security is more than a technical issue; it is a people, a process and an educational issue that must be addressed in its entirety.

Education is an important area of the security process. Security education should not only include raising awareness of the different types of technical attacks and social engineering attacks [8, 21], but it should also include information about the current environment. Employees should know with whom they should discuss security, how it fits into their everyday work environment (i.e. their development process), and the potential impact security has on the Web application solution that they are proposing / introducing into the organization.

4.2.3 Technology

Technological controls can be as granular as implementing proper authentication in order to preserve confidentiality, integrity and authentication through policy enforcement. Technological controls can also include the use of source control applications, the use of standardized application development software and up-to-date code libraries. Software can be used to analyze code to reduce the number of security vulnerabilities. Technology can also be utilized during application development by using project

management software and monitoring programs such as network intrusion detection and host based intrusion detection systems.

4.2.4 Process

The Gartner report refers to the process as "The newest and least-mature lens added to the resources of the information security officer"[7]. It goes on to say that "focus(ing) on process maturity can improve the quality of work and the efficiency with which it is accomplished (and that) the ability to translate efficiency into cost savings makes process maturity an easily justified investment"[7]. The process that an organization decides to implement is another form of control. This process can be in the form of a development process and a specific security process. It should be noted that there needs to be an application development process established either explicitly or implicitly within the organization. Without a development process there is serious potential for chaos. The results of the project then depend on the skill levels of the individuals involved.

The survey revealed three problems within the organization. The first problem is that the process is not used on all projects or is not followed properly for all projects. The second and more dangerous issue is that, realistically, the organization is operating two different approaches to application development at different levels within the organization. The danger with the issues is that security is implemented at the high level approach and ignored at the lower levels. This situation can mask security problems and make them more complicated to resolve. The third problem, the split process environment, naturally encourages a lack of consistency in the coding, documentation and delivery abilities between different development areas within the organization.

4.3 Security visibility

The third criterion is that security is visible throughout all areas of the development process. The organization's application development findings indicate that there is a problem with visibility due to the fact that, after a design has been formally approved, there is no verification that the implemented design matches the approved design. It also indicated that there is a much deeper security issue within the organization. The fact that the organization is operating two different development methodologies at different levels within the organization violates the visibility criteria. Security could potentially be implemented at the higher level and never filter down to the lower level. The security aspect of the organizational survey revealed that there are deficiencies in the areas of Business Analysis, Evaluation, Deployment, and Maintenance and Evolution.

Security should be visible in all steps of the development process if it is to be implemented with any success. This implies that the development process needs to be security focused. The term security focused translates into the use of effective and efficient designs, good coding practices, addressing security issues such as authentication and authorization issues, having specific security testing criteria, and acquiring feedback from the end user that is security specific. This means that the process encourages secure practices such as: acquiring specific security requirements, infrastructure re-use, re-usable components, coding standards, coding practices, end-to-end data security, secure designs, and takes into account security policies, procedures and standards.

Security solutions should also be confirmed with the end user. Does the solution meet the needs of the end user? If not, is the end

user circumventing the security measure? The survey indicates that there is a deficiency in the acquisition of end user feedback. This end user feedback deficiency is supported by other work in the same organization. [18]

4.4 Delivery

The goal of any development process should be to deliver a cohesive system, integrating business requirements / needs, software and security. This means that the security requirements of the business need to be identified as early as possible in the development process so that they can be incorporated into the design and the construction in order to produce secure software. The survey indicates that this does not happen within the organization. Security is lacking in the business analysis stage.

The incorporation of security into the development process should be as seamless as possible. The security that is implemented should meet the needs of the organization so that it adds value to the end product and to the overall business process. The application development area of the survey indicates that this criterion is not being met. The development process is not effective when considering time to market, rapid application deployment needs, introduction of new technology, and efficiency. Since security is not explicitly stated in the analysis phase of the process, the organization does not truly know if the business needs are being satisfied. To make matters worse, the survey revealed that budgets and time frames are often exceeded.

A metric system should also be developed that helps the organization determine the success of the development process security initiative. This should include issues ranging from general security education, to training, to monitoring and tracking all development bugs. This will help the organization determine if it is actually delivering a cohesive system that integrates the business, the software and the security perspectives.

4.5 Prompt, rigorous testing and evaluation

The development process should include rigorous end-user relevance testing and evaluation. Testing is critical to the success of an application. Testing should be conducted from a design and programming perspective using both automated and manual scripts, code reviews, and black, white and grey box testing. Testing should also take into consideration as much as is realistically achievable. This could include penetration testing and end user evaluation testing. End user testing translates into the process being accountable for the security requirements, the environment and the practicality of the solution from the end-user's perspective. Another sound testing practice is to bring in external testers to validate application security, when the risk is deemed appropriate for such an action.

The survey revealed, in the *Security Determination* section, that the process is not efficient in creating a situation where certain types of testing can occur on demand. Rigorous testing is a necessity in Web application development; however, the idea of possibly losing two weeks based on strict testing windows directly contradicts the Web application development need for short development life cycles. In a perfect world, testing should take place throughout the development life cycle; hence, utilizing short focused development cycles. However, this issue is dependent on the development life cycle the organization decides to implement as well as the cultural environment.

4.6 Trust and Accountability

The development process should encourage the development and maintainability of trust and accountability within the organization. Trust and accountability really make up the heart and soul of security.

4.6.1 Heart

Trust can be defined as "Firm reliance on the integrity, ability, or character of a person or thing"[5]. It is the foundation for a good relationship because it realistically adds value to the communication that takes place in the relationship[15]. Hence, Kaplan's reference to Gerick's explanation of trust is that "trust is not transitive, distributive, associative, or symmetric except in certain instances that are very narrowly defined"[15]. This information is of key importance to understanding the overall concept of trust. Establishing trust is the heart of security for without trust you can not rely on the information that is presented. A major component in gaining trust is to manage risk and then to implement appropriate controls, educate employees and monitor effectiveness[15]. A tried and true approach to identifying risk is a risk assessment initiative. Hence, trust should be identified in the risk assessment and mitigated in the design to establish and maintain trust. Since nothing is truly risk free, the goal is to mitigate the risk so that it is at an acceptable level. Hence, the development process has to take risk into consideration. This is typically done via a risk analysis. The earlier this is completed in the development process the better.

4.6.2 Soul

If the heart of security is trust, then the soul is accountability. Without accountability in a system there is no security just as, in life, without a soul there is no person. Accountability is critical to the enforcement of security. Individuals have to be successfully identified and authenticated in order to be held accountable for their actions through the use of logs and the effective implementation of access methodologies. The effective establishment of trust and realistic implementation of accountability controls should be visible within the organization's security policy, the application's design, coding practices, coding standards, application testing, and project feedback, as a project progresses through the application development life cycle.

5. CONCLUSION

A real world understanding of application security indicates that it is a multifaceted issue in an increasingly complex environment. This becomes especially apparent when examining Web facing applications. The need to address security in application development has increased over the past several years. However, one of the major challenges facing organizations in today's Web enabled environment is balancing technological needs with the business needs of the organization. Another potential challenge for organizations is structuring the overall development process so that there is not a general frustration within the organization in terms of overall process efficiency. A lack of process efficiency hinders aggressive Web development from a business perspective.

A lack of security integration and understanding of the application development process creates an environment that is conducive to fostering security deficiencies. The surveyed organization demonstrates this through its lack of security discussion in the beginning of the development process, lack of encouragement for

re-usable components, lack of follow up after design approval, and lack of employee understanding of the role security plays in the application development process. The results also indicate that there is a massive gap between the application development process and the implementation of security from an end-to-end perspective; therefore, it is vital to develop a security process that addresses security issues throughout that entire process. The conclusion derived from the survey's information is that the organization would benefit from a security education initiative in the technical arena, a well defined security process that details the individual responsible for each area, and tighter overall integration of security throughout the entire development process.

The results of the organizational survey, together with input from the relevant literature, support the need to establish a comprehensive Security Improvement Approach (SIA) for Web engineering development processes. Empirical evidence from the organizational survey supports the identification of six Security Criteria for Web Application Development (SCWAD):

1. Active organizational support for security in the Web development process
2. Proper Controls in the development environment
3. Security visibility throughout all areas of the development process
4. Delivery of a cohesive system, integrating business requirements, software and security
5. Prompt, rigorous testing and evaluation
6. Trust and Accountability

SCWAD provides an avenue for assessing existing Web Engineering processes and a guide to future Security Improvement Initiatives. This paper discusses the first component of an active case study. Future work will examine the application of a possible solution within the Fortune 500 organization. Future work will also need to take into consideration the application of SCWAD against existing application development methodologies to determine their security applicability.

6. REFERENCES

- [1] Berinato, S., *The Bugs Stop Here*, in *CIO*. 2003
- [2] Berinato, S., Global Security, The Global State of Information Security 2005. 13/10/2005. <http://www.cio.com/archive/091505/global.html>
- [3] Deloitte, *2005 Global Security Survey*. 2005, Deloitte Touché Tohmatsu: London. p. 1-44.
- [4] Deshpande, Y., Murugesan, S., Ginige, A., Hansen, S., Schwabe, D., Gaedke, M. and White, B., *Web Engineering*. *Journal of Web Engineering*, 2002. **1**(No. 1): p. 3-17.
- [5] Dictionary.com, Trust. 03/12/2005. <http://dictionary.reference.com/search?q=Trust>
- [6] Exler, R., Security and the Application Development Process. 22/01/2006. <http://www.csoonline.com/analyst/report3068.html>
- [7] Gartner Research, *Three Lenses Into Information Security*. 2006. p. 1-4.
- [8] Glisson, W. B. and Welland, R. *Web Development Evolution: The Assimilation of Web Engineering Security*. in *3rd Latin American Web Congress*. 2005. Buenos Aires - Argentina: IEEE CS Press.
- [9] Glisson, W. B., Glisson, L. M. and Welland, R. *Web Development Evolution: The Business Perspective on Security*. in *Thirty-Fifth Annual Western Decision Sciences Institute*. 2006. Hawaii: Western Decision Sciences Institute.
- [10] Glisson, W. B. and Welland, R., *Web Engineering Security (WES) Application Survey Technical Report*. 2006, University of Glasgow: Glasgow.
- [11] Gordon, L. A., Loeb, M. P., Lucyshyn, W. and Richardson, R., *2004 CSI/FBI Computer Crime Security Survey*. 2004, Computer Security Institute. p. 2-18.
- [12] Gordon, L. A., Loeb, M. P., Lucyshyn, W. and Richardson, R., *2005 CSI/FBI Computer Crime Survey*, in *Tenth Annual*. 2005, Computer Security Institute. p. 1-25.
- [13] Hansche, S., Berti, J. and Hare, C., *Official (ISC)2 Guide to the CISSP Exam*. 2004, Boca Raton: Auerbach Publications.
- [14] Hare, C., *Policy Development*, in *Information Security Management Handbook*, Tipton, H.F. and Krause, M., (eds). 2004, Auerbach Publications: Boca Raton. p. 925-943.
- [15] Kaplan, R., *A Matter of Trust*, in *Information Security Management Handbook*, Krause, H.F.T.a.M., (ed). 2004, Auerbach Publications: Boca Raton.
- [16] Lemos, R., Microsoft developers feel Windows pain. 23/10/2005. <http://news.com.com/2100-1001-832048.html>
- [17] McDonald, A., *The Agile Web Engineering (AWE) Process*, *Ph.D. Thesis*, in *Department of Computing Science*. 2004, University of Glasgow: Glasgow.
- [18] McDonald, A. and Welland, R., *Agile Web Engineering (AWE) Process: Perceptions within a Fortune 500 Financial Services Company*. *Journal of Web Engineering*, 2005. **4**(4): p. 283-312.
- [19] McGraw, G., *Software security*, in *IEEE Security & Privacy*. 2004. p. 80-83.
- [20] Mimoso, M. S., Top Web application security problems identified SearchSecurity.com. April 12, 2005. http://searchsecurity.techtarget.com/originalContent/0,289142,sid14_gci873823,00.html?NewsEL=9.25
- [21] Mitnick, K., *The art of deception : controlling the human element of security / Kevin D. Mitnick & William L. Simon*. 2002, Indianapolis, Ind.: Wiley. 352.
- [22] Organization for Internet Safety, Guidelines for Security Vulnerability Reporting and Response. <http://www.oisafety.org/guidelines/Guidelines%20for%20Security%20Vulnerability%20Reporting%20and%20Response%20V2.0.pdf>
- [23] Rakitin, S. R., *Software Verification and Validation: A practitioner's Guide*. 1997, Boston: Artech House.
- [24] Systems Security Engineering Capability Maturity Model (SSE-CMM) Project, *Systems Security Engineering - Capability Maturity Model (SSE-CMM) Model Description Document*. 2003, Carnegie Mellon University: Pennsylvania. p. 1-340.
- [25] Taylor, D. and McGraw, G., *Adopting a software security improvement program*, in *IEEE Security & Privacy*. 2005. p. 88-91.
- [26] Zelkowitz, M. V. and Wallace, D. R., *Experimental Models for Validating Technology*. *IEEE Computer*, 1998. **31**(5): p. 23-31.