ISSN: 2321-7782 (Online)
Volume 3, Issue 3, March 2015
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

Information Security requirement in Project Development Life Cycle

Manoj[1]
Deptt. of CSE
Sat Kabir Institute of Technology & Management (SKITM)
Bahadurgarh, Haryana, India

Shabnam Sangwan[2]
Deptt. of CSE
Sat Kabir Institute of Technology & Management (SKITM)
Bahadurgarh, Haryana, India

Abstract: ***The intent of this research is to attempt to determine how Information Security could be enhanced as a structured process and to develop an appropriate architectural framework and methodology that could enable integration of information security with enterprise project development life cycle (PDLC) processes. The research focuses on Information Security involvement in Software Project Development Life Cycle. Here we want to include Information Security considerations in project management to deliver more secure code/ applications in a more secure manner. As per industry best practices this practices has not been started at enterprise level. Project managers should consider the security requirements across project management life-cycle activities. We can achieve following parameters with the help of research:***

- ***To identify and quantify project specific security risks***
- ***To meet regulatory and contractual compliance***
- ***Highlight areas of non- compliance***
- ***To obtain assurance in areas of control inefficiencies and identification of remedial action***

Keywords: Information Security; Project Management Life-cycle (PDLC), Secure Development Policy (SDP), Standard Operating Procedure (SOP)

I. INTRODUCTION

The reliable Computing Security Development Lifecycle (SDLC) provides an example of a pragmatic way to incorporate security into development. The objective of the SDLC is not to overhaul an existing process totally but to add well-defined security checkpoints and security deliverables.

This main objective is to enhance an existing process by describing the security role for project checkpoints and deliverables, as well as discussing how security requirements affect project planning and monitoring.

To ensure that information security is designed and implemented within the development lifecycle of information systems..

II. LITERATURE REVIEW

The literature review for the research facilitates the analysis of literature materials. Examples of these include architecture frameworks, standards, policies, risk management etc.

International Standards Organization/International Electro technical Commission (ISO/IEC) 17799:2000 [1] provides procedures and code of practice for information security management in the enterprise. It outlines a general framework that provides a common basis for developing enterprise security standards and effective security management practices [4].

The control objectives and controls in ISO/IEC 17799:2005 [3] were intended to be implemented to meet the requirements identified by a risk assessment. It intended as a common basis and practical guideline for developing organizational security standards and effective security management practices, and to help build confidence in inter-organizational activities

Enterprises like HCL Technologies use a holistic approach to Information Security Management, and establish an Information Security management system. This system would integrate policies, standards, guidelines, code-of-practice, technology, human issues, legal, and ethical issues. This means using a process model approach to manage information security. The authors propose “process security” and “product security” in information security management. In “process security”, the focus would be on planning. In “product security”, the focus would be on the use of certified software products in the IT infrastructure in order to establish and maintain information security.

Jan H.P. Eloff and Mariki Eloff [4] provided a consolidated approach to the evaluation of IS management, in terms of which full cognisance will be taken of both these perspectives.

Later M.M Eloff, S.H von Solms [5] worked on product and system evaluation for procedural perspective.

Russell L. Jones and Abhinav Rastogi [6] elaborated in their research that to meet future needs, opportunities, and threats associated with information security, security needs to be “baked in” to the overall systems development life-cycle process.

III. Purpose And Scope

The intent of this research is on involvement or requirement of Information security while developing a project or product e.g. .Net, Java development projects. The research also aims to develop an appropriate architectural framework and methodology that could enable integration of information security management with Software development life cycle processes. In Proceedings of the Annual ISO Conference, Control A.6.1.5 project management is embedded in ISO 27001:2013 standard, which is internationally accepted by all enterprises

The results of this research would be important to any enterprises or organization with a need for a secure business environment, especially when they are doing Software development, maintenance, testing projects. The research results will also be important to individuals responsible for managing information security in their projects, as well as to Customers, partners because of their increased statutory responsibilities to secure various types of information in their organizations.will be removed from papers during the processing of papers for publication. If you need to refer to an Internet email address or URL in your paper, you must type out the address or URL fully in Regular font.

IV. Methodology

For enterprises moving to make security a higher priority, project teams or managers need to implement or address the following:

A. Secure development Policy (SDP)

Secure development is a requirement to build up a secure service, architecture, software and system. Within a secure development policy, the following aspects should be put under consideration:

a) Security of the development environment;

b) Guidance on the security in the software development lifecycle;

c) Secure coding guidelines for each programming language used;

d) Security requirements in the design phase;

e) Security checkpoints within the project milestones;

f) secure repositories; version control, access control

g) Developers' capability of avoiding, finding and fixing vulnerabilities.

An Organization Information Security Governance Team need to create or document Secure Development Policy (This may be a part of corporate Information Security Policy). If there were no formal documented Policy, then organization personnel at any level would have no guidance on how to make decisions during entire SDLC. This helps employees to initiate actions and take responsibility without constant reference to management. Increase the accountability of business or organization's and its staff.

In reality, however, the existence of Policy provides many benefits provided they are written well and kept up to date. This Policy need to be reviewed at least once in a Year or update in case of any changes in Policy. After review and changes (if any), this should be approved by Chief Information Officer (CIO), Chief Information Security Officer (CISO) or Department Heads only.

B. Standard Operating Procedures (SOP)

Operating procedures shall be documented and made available to all developers/ users who need them. SOP should be in place so that development team should know the security requirements on each and every phase during project phases of development life cycle. All step by step instructions should be incorporated into the document with individual's roles & responsibilities.

The approved procedure is documented in a format that is easy to follow and reduces the chance of errors being made. The idea behind it is to reduce the possibility of human error and to provide guidelines for employees to follow.

C. Segregation of Development, testing and production environments:

Development, testing and production environment shall be separated to reduce risks and threats of unauthorized access or changes to the operational environment.

The intent of this requirement is to ensure that development/ test functions are separated from production functions. For example, a developer may use an administrator-level account with elevated privileges for use in the development environment, and have a separate account with user-level access to the production environment.

In environments where one individual performs multiple roles (for example application development and implementing updates to production systems), duties should be assigned such that no one individual has end-to-end control of a process without an independent checkpoint. For example, assign responsibility for development, authorization and monitoring to separate individuals.

Reducing the number of personnel with access to the production environment minimizes risk and helps ensure that access is limited to those individuals with a business need to know.

Organization IT Services or Technology (Network & Server) team is responsible for segregation of all environments i.e. Development, Testing and Production and access control as per business needs. Network team recommends separate VLAN for the environments and Server team is responsible for Server deployment in Data Center or Server Room. Security processes such as Change Management, Secure guidelines as per standard operating environment to be followed as per Industry best practices.

D. Protection from Malware

Malware is "malicious software." This is software that is specifically designed to gain access or damage a computer without the knowledge of the owner. There are various types of malware including spyware, key loggers, true viruses, worms, or any type of malicious code that infiltrates a computer. Generally, software is considered malware based on the intent of the

creator rather than its actual features. Malware creation is on the rise due to the sheer volume of new types created daily and the lure of money that can be made through organized internet crime. Malware was originally created as experiments and pranks, but eventually led to vandalism and destruction of targeted machines. Today, much of malware is created for profit through forced advertising (adware), stealing sensitive or confidential information (spyware), and spreading email spam. Various factors can make computers more vulnerable to malware attacks, including defects in the operating system design, having all of the computers on a network run the same OS, giving users to much permissions or just using the Windows OS (due to its popularity, it gets the most malware written for it).

During Software Development Life Cycle (SDLC), we need to ensure that information and information processing facilities are protected against malwares. Detection, prevention and recovery controls shall be implemented to protect against malwares.

The best protection from malware continues to be the usual advice: be careful about what email attachments you open, be cautious when surfing and stay away from suspicious websites, and install and maintain an updated, quality antivirus programs such as : Symantec endpoint protection, McAfee Computer security etc.

E. Change Management

A proper *Change Management Process* should be followed in case of any changes during development up to till production.

Threats may occur sometime in case of false changes are made without any testing (or staging). The development environment requires a level of security commensurate with the planned security level of the software product being produced. Appropriate controls and configuration management of the development artifacts are essential. There may be specific tools required, such as for static code analysis, to aid the production or testing of secure software.

Change is a necessary part overall SDLC process. However, many leaders manage change poorly, causing distrust or confusion for their employees and clients. Organization responsible team should write a change management plan, leaders intentionally design a process that helps everyone know what needs to change, why it needs to change, and how to go about making the change take place smoothly. For more details, employees refer Change control procedures - 12.5.1 of ISO 27002:2013 (Code of Practice).

F. Risk Assessment

Security Risks, threats and vulnerabilities are the potential attackers and are described in terms of (employee, business partner, contractor or supplier) with an objective (financial gain, project code, company's IPR or obtaining proprietary corporate information, disabling essential business systems), and with a set of resources (personnel, computing software, tools, hardware, skill, knowledge of internal systems).

A risk assessment explores how a component could be exploited by the identified threats and vulnerabilities (i.e., what could go wrong) and analyzes the possible responses to such attacks. The response options for a risk are to (a) mitigate (reduce probability of event, reduce impact, improve recovery), (b) transfer (insurance, contracted agreements), (c) ignore (for low impact and highly unlikely threats), or (d) avoid, which may require changes in requirements. The factors involved with a risk assessment that is done early in the development process are predominantly business rather than technical. Project management needs to ensure stakeholder participation in such activities. The attack patterns would be rather abstract for a preliminary system risk assessment and would become more detailed as the software architecture and detailed design are created.

Risk management process to be followed by organization where Information Security experts need to identify, assess, and prioritize the risks followed by coordinated and economical application of resources to minimize, monitor, and control the probability and/or impact of unfortunate events or to maximize the realization of opportunities.

Refer - ISO 31000: 2009 'Risk management - Principles and guidelines'

G. Activities required to complete deliverables:

Regulatory or contractual compliance may require demonstration that the software provides the necessary controls for accessing the information (i.e., the production of an assurance case). Security governance typically increases the complexity for meeting security requirements. For example, business process compliance may require showing that the composition and interactions of multiple applications maintain the required controls and feedback.

Regulatory compliance as per ISO/IEC 27001:2013 (Information Security standard) Contractual compliances requirements are set by Customers only and vendors or suppliers should adhere those guidelines.

Delivering "Secure" software requires demonstrating that the desired level of assurance has been achieved. While demonstrating that a system provides the required functionality is an essential aspect of software assurance, software security assurance depends more on demonstrating what a system does not do. An improper input leads to a system failure or enable an attacker to bypass authentication.

The production of such an assurance case must be planned and managed. An assurance case provides an argument for how the software meets an identified threat. That argument typically is based on assumptions about how the software behaves under certain operating conditions. Hence, an early step in building an assurance case is to provide evidence that software behavior satisfies the assumptions of the assurance argument. The production of an assurance case is an incremental activity. The assurance case should describe the architecture's role for meeting security requirements, and the architectural risk assessment and analysis should provide evidence that the architecture satisfies those requirements.

Syntactic analysis of the source code reduces the probability of coding errors that might lead to security vulnerability. Risk-based testing can target the components and interfaces that are most likely to lead to a system compromise.

H. Restrictions on software/ tools installation

Procedures shall be implemented to control the installation of tools/ software's on development, testing and production systems.

Restrictions on software installation - The organization should define and enforce strict policy on which types of software users may install. The principle of least privilege should be applied. If granted certain privileges, users may have the ability to install software. The organization should identify what types of software installations are permitted (e.g. updates and security patches to existing software) and what types of installations are prohibited (e.g. software that is only for personal use and software whose pedigree with regard to being potentially malicious is unknown or suspect). These privileges should be granted having regard to the roles of the users concerned.

Uncontrolled installation of software on computing devices can lead to introducing vulnerabilities and then to information leakage, loss of integrity or other information security incidents, or to violation of intellectual property rights.

Note: Software licensing agreements that are such that organizations may become liable for licensing for client software on workstations owned privately by employees or external party users;

I. System and application (code level) access control

Access to information and application system functions should be restricted in accordance with the access control policy. Restrictions to access should be based on individual business application requirements and in accordance with the defined access control policy.

a) The following should be considered in order to support access restriction requirements:

b) Providing menus to control access to application system functions;

c) Controlling which data can be accessed by a particular user;

d) Controlling the access rights of users, e.g. read, write, delete and execute;

e) Controlling the access rights of other applications;

f) Limiting the information contained in outputs;

g) Providing physical or logical access controls for the isolation of sensitive applications, application data, or systems.

h) Secure log-on procedures

J. Use of secret authentication information/source code:

Users should be required to follow the secure coding guidelines/ practices in the use of secret authentication information. All users should be advised to:

a) Keep secret code/ authentication information confidential, ensuring that it is not divulged to any other parties, including people of authority

b) Avoid keeping a source code/ important records (e.g. on paper, hand-held device)

c) Information, unless this can be stored securely and the method of storing has been approved (e.g. password vault)

d) Change secret authentication information whenever there is any indication of its possible compromise

e) When passwords are used as secret authentication information, select quality passwords with sufficient minimum length.

f) Ensure proper protection of passwords when passwords are used as secret authentication.

g) Information in automated log-on procedures and are stored.

h) Not use the same secret authentication information for business and non-business purposes.

K. Facilities and Staffing

Security expertise on most projects is limited and may be an internal or a contracted service. The allocation of that resource is often difficult even when security activity is limited to networks, authentication, and access control, but when security has to be incorporated into application development, that expertise is spread much thinner. An increase in the level of assurance can significantly affect the both the security and software engineering expertise required.

For this discussion, we divide security expertise into two categories. One category consists of knowledge of security functionality such as the specification and implementation of access control, authentication, and encryption functions. Such security functionality may be encapsulated in the system infrastructure. The second category of expertise consists of the skills to identify and mitigate exploitable system vulnerabilities. Historically, a significant number of the vulnerabilities that lead to a security failure were created by application errors and not by failures with the security infrastructure. Vulnerabilities may be in the least exercised parts of the system and depend on pathological aspects of the interface. Such vulnerabilities may be missed by application development teams, who normally concentrate on the core functionality.

The security functionality for authentication, authorization, and encryption is typically composed of commercially supplied components that can be tailored for a specific operating environment. Those components must have the required assurance level. It would not be surprising to find the security knowledge associated with the first category to be concentrated within a few teams. The security specialists associated with that infrastructure should be aware of the security issues associated with development and project management. Unfortunately, application development teams rarely have the necessary security expertise. The resources in the second security knowledge category must be spread across multiple development efforts.

Tasks such as risk assessments, code reviews, threat modeling, Vulnerability assessment, require security expertise. On the other hand, there are security improvement practices that can be implemented without requiring extensive security experience. For example, although security knowledge may be necessary to configure a tool for the static analysis of the source code, the use of such a tool does not require a security background. (See the Code Analysis Tools content area.) Testing provides a second example. Penetration testing is often part of an acceptance test or certification process. Penetration testing might be implemented by what is called a red team: security experts who attempt to breach the system defenses. Fuzz testing is a simple form of penetration testing that finds software defects by feeding purposely invalid and ill-formed data as input to program interfaces. Fuzz testing does not replace the need for testing that targets explicit security risks, but it is an example of an approach that can be used without detailed knowledge of security vulnerabilities.

L. Project and Product Risks

Poor management of requirements scope is another frequent cause for project failure. Scope management is particularly important where the learning curve is a necessity because of the immaturity of the business usage or the supporting technology. Business integration requirements are pushing the connectivity of networked information systems beyond an organization's IT systems. Meeting business requirements may depend on using relatively new protocols such as those for Web Services. Those protocols are currently a moving target, as they continue to be revised to reflect the experiences of early adopters. Best practices in this context have short lives, and the lack of well-defined and proven practices adversely affects planning. Plans for these circumstances might include a prototype or use of an iterative or incremental approach.

Security mechanisms that mitigate a specific risk may create additional ones. For example, security requirements for managing identity for a large distributed system might be met by implementing authentication and authorization as infrastructure services shared by all applications, but the aggregation of authentication and authorization mechanisms into a shared service makes that service a single point of failure and a possible attack target. Such design decisions should involve a risk assessment to identify any new threats that require mediation, as well as the analysis of the operational costs after the system is deployed.

V. CONCLUSION

The potential outcome and value of validation of the research proposition could be an approach to implement an information security while developing a project or product and their applications. This approach would include an architectural framework and methodology, a security policy framework, and a supporting process model that could enable integration of information security with enterprise project life cycle processes. A Secure code guidelines/ checklists to be performed at enterprise or organization level to meet all security requirements. Apart from this we will conduct application Vulnerability Assessment and Penetration Testing to check all threats and risks associated with in the environment. An objective for the Chief Information Security Officer (CISO) of Organizations is to "raise the bar" for component Software assurance by integrating assurance into the development processes is taken in account. The production of an assurance case can serve as an integrating mechanism by identifying threats and desired responses and then tracing and refining the threats and responses during development.

VI. FUTURE SCOPE

Though the methodologies has been successfully demonstrated by use of different result vectors and threat vectors, but there have been some limitations in the research work carried out in this thesis. One of the limitations, in the proposed framework is automatic integrated tool involvement. This portion of the framework requires human intervention who can classify all security risks, threats and vulnerabilities associated within projects starting from initial stage to last stage. In future this research effort can be put on to a dedicated Security-Project Management tool. Thus, this portion of the work (included all methodologies) can also be automated. Also from compliance point of view we can automate control reference of ISO

27001:2013. Hence, there is a lot of future scope of the research work to be carried out in this vital area of great significance to mankind.

ACKNOWLEDGMENT

I would like to thanks my worthy guide Ms. Shabnam Sangwan, who suggested me to work and research Information Security in Project Development Life Cycle. Her recommendations, innovative ideas and constructive criticism contributed to make the success of this report. Her numerous suggestions, comments, and advice have made this entire paper possible.

References

1. ISO/IEC. ISO/IEC 17799: Information Technology–Code of Practice for Information Security Management. 2000.
2. National Institute of Standards and Technology (NIST). "International Standard ISO/IEC 17799:2000 Code of Practice for Information Security Management – Frequently Asked Questions." November 2002.
3. ISO 31000 : 2009 'Risk management — Principles and guidelines'
4. ISO/IEC 27001:2013 - Information technology -- Security techniques -- Code of practice for information security management, 2013.
5. Jan H.P. Eloff and Mariki Eloff, "Information Security in Management-A new Paradigm", Annual Research Conference of the South African institute of computer scientists and information", SAICSIT '03, 2003.
6. M.M Eloff, S.H von Solms, Information Security Management: An Approach to Combine Process Certification And Product Evaluation, Computers & Security, Volume 19, Issue 8, 1 December 2000.
7. Russell L. Jones and Abhinav Rastogi, "Secure Coding: Building Security into the Software Development Life Cycle", Information Systems Security,Volume 13, Issue 5, 2004
8. Enrico Vezzetti, Maria Grazia Violante, Federica Marcolin "A benchmarking framework for product lifecycle management (PLM) maturity models", The International Journal of Advanced Manufacturing Technology, 2013.
9. http://projects.webappsec.org/w/page/13246988/Web%20Application%20Security%20Scanner%20List
10. http://resources.infosecinstitute.com/14-popular-web-application-vulnerability-scanners/
11. Mead, N.R., Viswanathan, V., Padmanabhan, D., and Raveendran, A., Incorporating Security Quality Requirements Engineering (SQUARE) into Standard Life-Cycle Models. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2008.
12. A book on "The Security Development Lifecycle" by Michael Howard and Steve Lipner, Microsoft Press, May 2006, http://www. microsoft.com /MSPress/books/8753.aspx.
13. Mellado,Fernandez and piattini, "A Common Criteria Based Security Requirements Engineering Process for Development of Secure Information Systems" , Computer Standards & Interfaces, Volume 29, Issue 2, February 2007.

AUTHOR(S) PROFILE

Manoj, received the B.E degree in Computer Science & Engineering from Bhagwan Mahavir Institute of Technology & Management (BMIET), Sonepat affiliated to Maharshi Dayanand University, Rohtak (Haryana) and pursuing M.Tech (2013 to 2015 batch) from Sat Kabir Institute of Technology and Management (SKITM), Bahadurgarh affiliated to Maharshi Dayanand University, Rohtak (Haryana). Currently I am doing research on Information & Data Security – A Serious concern to IT Organizations. I am thankful to Management and staff of college who is supporting me during my entire research/ Thesis work.

Shabnam Sangwan, received the B.Tech degree in Computer Science & Engineering from Maharaja Surajmal Institute of Technology (MSIT), affiliated to Guru Gobind Singh Indraprastha University, New Delhi and M.Tech degree in Computer Science & Engineering from PDM college of Engineering, affiliated to Maharshi Dayanand University, Rohtak (Haryana) in 2011 and 2013 batch respectively. She is presently working in Sat Kabir Institute of Technology and Management (SKITM), Bahadurgarh, Haryana, India and had associated with PDM polytechnic, Bahadurgarh earlier to this.

