A Study of Software Development Prevention Checklist for Security Assurance

Syed Tassuduq Hussain[1], Muhammad Kashif Siddiqui[2]

[1]*MSCS SZABIST, Karachi*

Sthussain81@yahoo.com

[2]*National Bank of Pakistan, Karachi*

kashif.siddiqui@nbp.com.pk

Abstract: *The security is a serious concern when developing enterprise level application and to cope up with these challenges companies reevaluating their practices that help them to construct secure application.*

This study will present Software Development prevention checklists for security that assure the security concern are being treated very comprehensively. This report also classifies security assurance methods, techniques and suggestions.

Keywords: *Security Review, Security Walkthrough, Security Inspection, Security Assurance*

1. INTRODUCTION

The security always been a serious concern when developing Enterprise level application or larger Information systems and to cope up with these challenges companies reevaluating their practices that help them to construct secure application.

Security assurance basically provides a basis through which the application can be assure for attack and illegal access. Ideally the application must be such liable or authoritative that it can not be accessed by any unauthorized user or code.

In Software development life cycle (SDLC), security is not very well versed as compare to other properties of the software. Not even software engineer in a condition to claim that the particular set of activities can make

application secure. Software security has its dynamic property. It looks secure for particular environment but change of environment can transform the level of confidence. The process immaturity passes its burden to security testing and as a result the fault and uncounted defect will produce that plays major part in software failure [1].

In August 1999, the US Congress General Accounting Office (GAO, now the Government Accountability Office) published a report to the Secretary of Defense entitled DoD Information Security: Serious Weaknesses Continue to Place Defense Operations at Risk. [1]

In the area of "Application Software Development and Change Controls," GAO reported that—Structured methodologies for designing, developing, and maintaining applications were inadequate or nonexistent. There was no requirement for users to document the planning and review of application changes and to test them to ensure that the system functioned as intended.

Also, application programs were not adequately documented with a full description of the purpose and function of each module, which increases the risk that a developer making program changes will unknowingly subvert new or existing application controls found that application programmers, users, and computer operators had direct access to production resources, increasing the risk that unauthorized changes to production programs and data could be made and not detected.

Different international organizations define security assurances that are define as under:

1.1 CNSS Definition

The Committee on National Security Systems (CNSS) defines software security assurance as:

"The level of confidence that software is free from vulnerabilities, regardless of whether they are intentionally designed into the software or accidentally inserted later in its life cycle, and that the software functions in the intended manner". [2]

1.2 DoD Definition

The Department of Defense's (DoD) defines software security assurance as:
"The level of confidence that software functions as intended and is free of vulnerabilities, either intentionally or unintentionally designed or inserted as part of the software". [3]

1.3 NASA Definition

The National Aeronautics and Space Administration (NASA) define software security assurance as:
"The planned and systematic set of activities that ensure that software processes and products conform to requirements, standards, and procedures". [4]

2. SECURITY REQUIREMENT

Security Requirements plays an important role for developing any enterprise level application. It is well recognize in industry to be an essential part to the success of any crucial development project. Various studies have shown that if requirement are not properly handled it cost 10 to 200 times more to correct once done [Boehm 88, McConnell 01] while other studies have shown that reworking on requirements defects costs 40 to 50 percent of total project effort [Jones 86], and the percentage of defects originating during requirements engineering is estimated at more than 50 percent [Wiegers 01].

A recent study found that the return on investment when security analysis and secure engineering practices are introduced early in the development cycle ranges from 12 to 21 percent, with the highest rate of return occurring when the analysis is performed during application design [Soo Hoo 01]. NIST reports that software that is faulty in security and reliability costs the economy $59.5 billion annually in breakdowns and repairs [NIST 02].

The costs of poor security requirements show that there would be a high value to even a small improvement in this area. By the time that an application is fielded and in its operational environment, it is very difficult and expensive to significantly improve its security. [5]

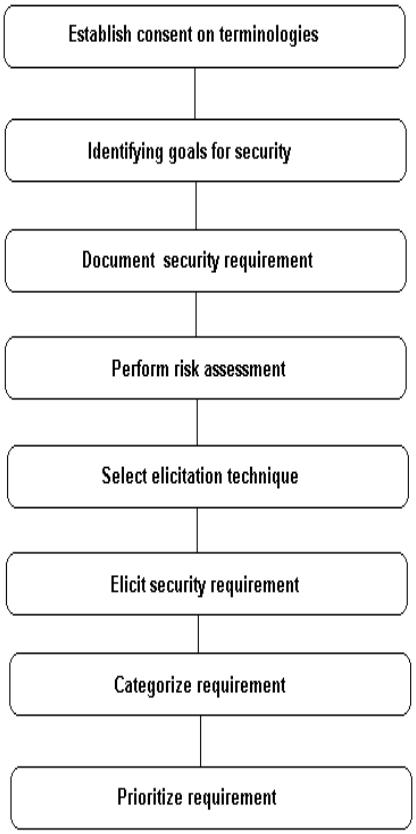

Figure 01: Workflow of Security requirement

2.1 Establish consent on terminologies

For effective and unambiguous communication the requirement engineer and other stakeholder must agree on the basic terminology because the difference in terminology can create major flaws. The following are some important point that needs to be considered at this level.
All relevant terminologies that are used in the project must describe properly with suggested definition. The suggested definition must be very clear and complete.

If there is any external source that is define, mention it properly.
All sources should be provided to all stakeholders to review.

All stakeholders must come to same point after the exchanging of suggested definition. Document all the terminologies and share it after finalization. Established point of contract between the stakeholder and the requirement engineer.

2.2 Identifying goals for security

Identifying goals for security is very important factor for taking requirement. The purpose of this stage is to identifying goals for security for example the stakeholder want to keep detail record of the modification history for financial activities but not for the Sales department. The following are some important point that needs to be considered at this level.

Facilitate the brainstorm session by the stakeholders, emphasizing the importance of creating a single business goal, followed by several security goals that support it.

Review the stakeholders' business and security goals, providing any feedback on scope, level of detail, and relevance to the business goal of the project.
Document and share the finalized business goal and corresponding security goals.

2.3 Document security requirement

The following are some important point that needs to be considered at this level. Work with the stakeholders and client organization to identify and collect as many artifacts as possible and based on these artifact document all security requirement as possible. Verify the accuracy and completeness of all artifacts.

2.4 Perform risk assessment

The following are some important point that needs to be considered at this level. Facilitate the completion of a structured risk assessment, likely performed by an external or internal risk expert. Review the results of the risk assessment and share them with stakeholders.

2.5 Select elicitation technique

The following are some important point that needs to be considered at this level. Select an elicitation technique that is appropriate for the number and expertise of

stakeholders, size and scope of the project, and expertise of the requirements engineering team.

Document the rationale for the choice and make necessary preparations to execute the technique.

2.6 Elicit security requirement

The following are some important point that needs to be considered at this level.

Execute the elicitation technique that were selected previously document the requirements as they are collected.

2.7 Requirement categorization

The following are some important point that needs to be considered at this level. Provide a baseline set of categories to system, software and architecture level. The team may have to suggest alternative categories, depending on the client project. Facilitate the stakeholders' categorization process.

2.8 Prioritize requirement

The following are some important point that needs to be considered at this level. Prioritize the security requirements using the risk assessment and categorization results as a basis for decision making

3. Security Review

In the planning phase of the project, Software Security Assurance team plans for the number of formal reviews (inspections) required for the life cycle of project with the consent of other stakeholder.

Walkthroughs may or may not be planned at the start of the project and can be conducted when concerned role/person feels the requirement of walkthroughs. Following are the processes of walkthroughs and inspections:

3.1 Walkthrough

The objective of security walkthrough is to finding problems, discussing alternative solution and focusing on demonstration how work product meet all the requirement. The following is the picture having detail view of this process.

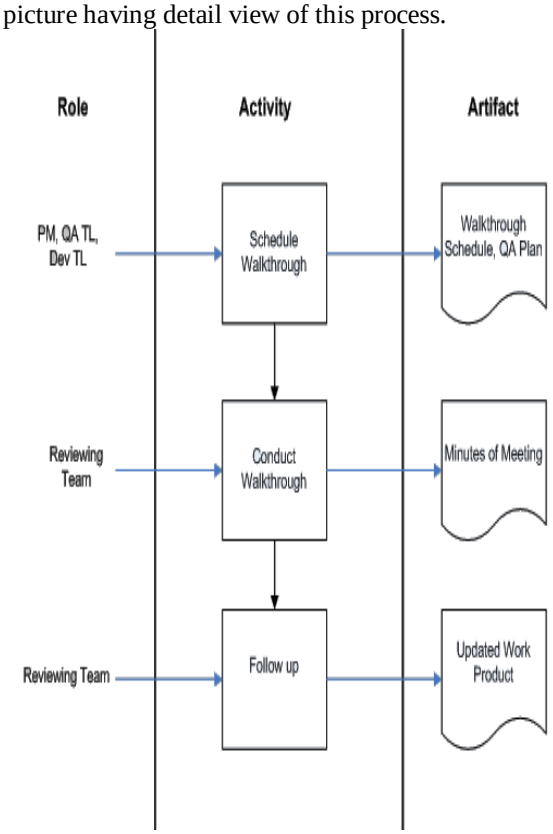

Figure 02: Workflow of Security walkthrough

The following is the process of walkthrough.

Concerned stakeholder selects review participants, obtains their agreement to participate, and schedules a walkthrough meeting.

Concerned stakeholder plans the walkthrough in QA Plan as required. Walkthrough may not be planned in QA plan and can be conducted on need basis
Author distributes work product to reviewers prior to the meeting if possible.

Author describes the work product to the reviewers during the meeting in any appropriate way. Reviewers identify possible defects, and improvements/suggestions.

Based on identified defects or suggestions, author performs necessary rework if possible during the meeting.

Author prepares minutes of meeting and sends it to all relevant stakeholders along with the agreed upon follow up date, if any via email.

Author makes required changes in work product
Reviewers follow up walkthrough action items on agreed upon date to ensure all decided changes are made.

3.2 Inspection

An inspection is a formal, rigorous, in-depth group review designed to identify problems as close to their point of origin as possible. Inspection is a recognized industry best practice to improve the quality of a product and to improve productivity.

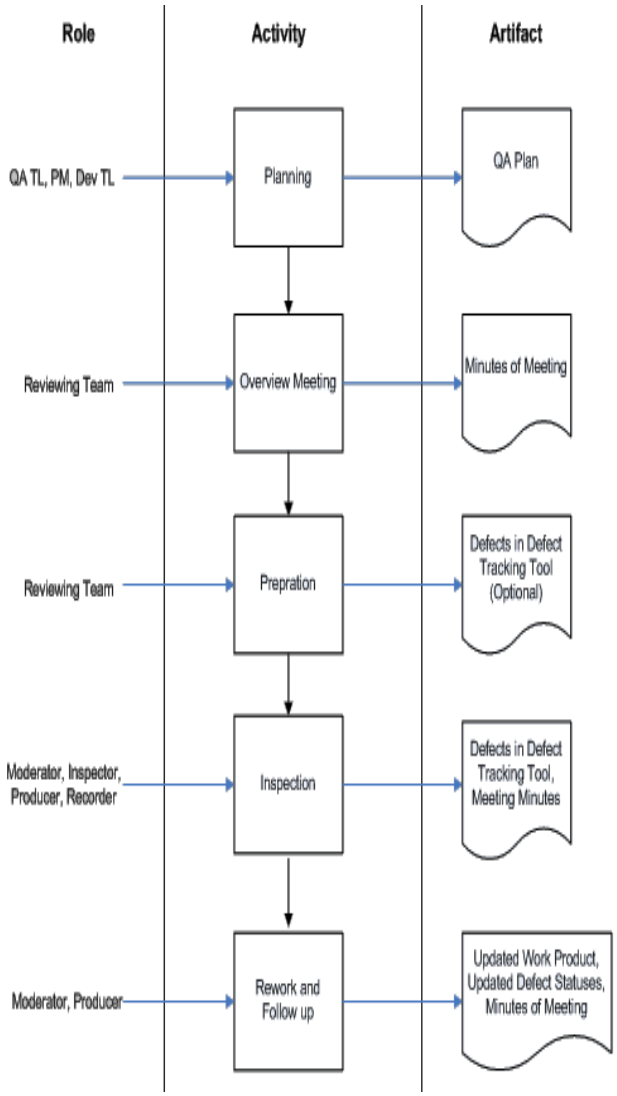

Figure 03: Workflow of Security Inspection

3.2.1 Planning

Concerned role/person (e.g. PM, QA TL, Dev TL) identifies the work product to be inspected and determines if the work product is ready to be inspected.
Concerned role/person (e.g. PM, QA TL, Dev TL) selects the moderator.

3.2.3
Once the moderator has been selected and has accepted the 3.2.4
assignment, project manager and moderator select inspection team.

Moderator assigns the roles of Inspector, Producer, Reader, Recorder and Observer to selected inspection team members in coordination with PM.

The moderator obtains commitment from each team member to participate. This commitment means the person agrees to spend the time required to perform his or her assigned role on the team. In some cases, approval from the team member's 3.2.5
supervisor or manager may be required. 3.2.6

Moderator ensures the availability of required facilities and makes necessary reservations.
The moderator and the producer decide if an overview meeting is required based on the inspection team's familiarity with the work product and the complexity of the work product being inspected.

The moderator and the producer identify the review materials required for the inspection. The moderator ensures that the review materials are distributed. The review material includes the specification of the wok product and relevant checklists and standards. The specifications of the work product are generally the outputs of the previous phases and are needed to verify the work product to be inspected.

The moderator schedules meetings and distributes review materials. The moderator communicates the date, time, and location of the meetings to the inspection team and gets their commitment. If an overview meeting is held, the moderator can distribute the review materials during that meeting.

3.2.2 Overview meeting

The moderator opens the meeting and describes the review objectives. The moderator distributes the work product and the review materials. The producer describes the information contained in the review materials. The producer provides overview of the work product and also mentions any special considerations, assumptions, constraints and areas that need to be described in advance.

Team members ask questions to facilitate their understanding of the work product and the information in the review materials. Recorder records minutes of meeting and sends the document to relevant roles/persons.

Preparation

Inspectors review checklists and internal standards and conventions before reviewing work product to create material list of things to become more familiar with review materials and work product. While preparation, record any obvious defects. Ideally the preparation for review shall be done in one continuous time span. The producer may be informed about the logged defects in defect tracking tool so that he can become familiar with outputs of each reviewer and prepare for the final meeting.

Inspection meeting

The moderator opens the meeting. The moderator determines if the inspectors are prepared. If moderator identifies that the team is not adequately prepared, the moderator postpones the meeting. If the moderator is satisfied that the team is adequately prepared, the inspection begins. Reader starts by paraphrasing the first chunk of information from the work product (or present the material by any other convenient method).

During the meeting, if an inspector has previously identified any defect or finds a new defect, he or she raises the point. The defect is discussed among the team. The producer reviews the defect under discussion and either clarifies why it is not a defect or accepts it as a defect. Sources of defects are also identified. Recorder notes down all the defects and their sources. After the reader has completed the entire work product, the moderator asks the recorder to read back all the noted defects to ensure that they were recorded correctly.

On the basis of defects' severity, the team decides about re-inspection. If another meeting or re-inspection is required, the moderator schedules it. Recommendation regarding reviews in the next stage can be made.

The moderator adjourns the meeting. Recorder logs all the defects and their sources in defect tracking tool and assigns them to producer. Minutes of meeting are prepared and sent by the recorder to the relevant roles/persons as required.

3.2.7 Rework and follow-up

The producer and the moderator agree on the schedule for completing corrective action. The producer fixes the defects

identified by the inspection team. If the inspectors are assigned some issues, they must investigate those issues and submit the result to the moderator and producer. When all rework has been completed, the moderator verifies the rework and updates the defects status in defect tracking tool or reschedules a follow-up inspection meeting (if defects not fixed), as determined by the team.

REFERENCES

[1] Goertzel, Thomas, Robert, Elaine. "Software Security Assurance: A state-of-the-Art report". July 31 2007.

[2] Committee on National Security Systems, National Information Assurance (IA) Glossary, CNSS instruction No. 4009 (revised June 2006).
Available from: http://www.cnss.gov/Assets/pdf/cnssi_4009.pdf

[3] Mitchell Komaroff (ASD/NII) and Kristin Baldwin (OSD/AT&L), DoD Software Assurance Initiative (September 13, 2005).
Available from: https://acc.dau.mil/CommunityBrowser.aspx?id=25749

[4] National Aeronautics and Space Administration (NASA), Software Assurance Standard, Standard No. NASA-STD-2201-93 (Washington, DC: NASA, November 10, 1992).
Available from: http://satc.gsfc.nasa.gov/assure/assurepage.html

[5]Nancy R. Mead, Eric D. Hough, Theodore R. Stehney security Quality Requirements Engineering (SQUARE) Methodology November 2005