Security Requirements Engineering Using Outsourcing

A. Meligy, H. Diab, M. Torky
Faculty of Science
Menofia University
Egypt

ABSTRACT: *Now days several organizations started to build their security system during SDLC depending on an Outsourcing companies. The purpose of this study is to show how the outsourcer represented in Managed Security Service Provider (MSSP) can define and specify security requirements during software requirement engineering process. This paper introduced a new model that aims to define and specify security requirements during software requirement engineering using MSSP who can be considered as a new security member of the development team. Our new model added major security activities which should be performed via MSSP during software requirement engineering process. We analyzed our new model using CMMI and estimated it using COCOMO II-Early Design Model. We concluded that, using our new model will increase the security of customer's software system from the beginning of SDLC, and involving MSSP in the process will save security effort,Cost and Time.*

Keywords: Outsourcing, MSSP, Security Requirements, OSRE Model, CMMI and COCOMOII-Early Design Model

Received: 11 June 2012, Revised 29 July 2012, Accepted 3 August 2012

1. Introduction

The problem of producing dependable software system require from system engineers to consider security issues during all phases of software development life cycle (SDLC) [2]. So security engineering is therefore an increasingly important aspect of software development life cycle (SDLC) which involves some of security practices and activities associated with the basic activities of software engineering process. Hence, considering security issues especially during the early stages of software development life cycle (SDLC) will prevent security vulnerabilities from appearing in software system after it deployed and configured with specific work environment [1]. Nowdays several organizations started to depend on Outsourcing companies [7] to perform security engineering activities instead of depending on some of local security specialists. The transition to Outsourcing because of its advantages such that better security performance, updated technologies, cost effectiveness and well trained specialists, etc. This paper aims to develop a new methodology represented in new model which handle defining and specifying security requirements during software requirement engineering process. The novelty in our proposed model is, the core security activities will be the responsibility of Outsourcer which we usually called *Managed Security Service Provider* (*MSSP*), not the job of software engineers, also the customer will share the MSSP in specifying his security requirements. By this transform in our study, the software engineers will focus only on specifying the general software requirements and transfer specifying security requirements to the new security member (i.e.MSSP). Before we explains the model in details, we should assume that, the model focuses only specifying security requirements, and the leading role will be in the hand of MSSP. a model expected outcome include: (A) the feasibility study will involve the security considerations (B) software requirement elicitations will involve eliciting Security requirements via MSSP. (C) customer can share MSSP in security requirements elicitation and validation.

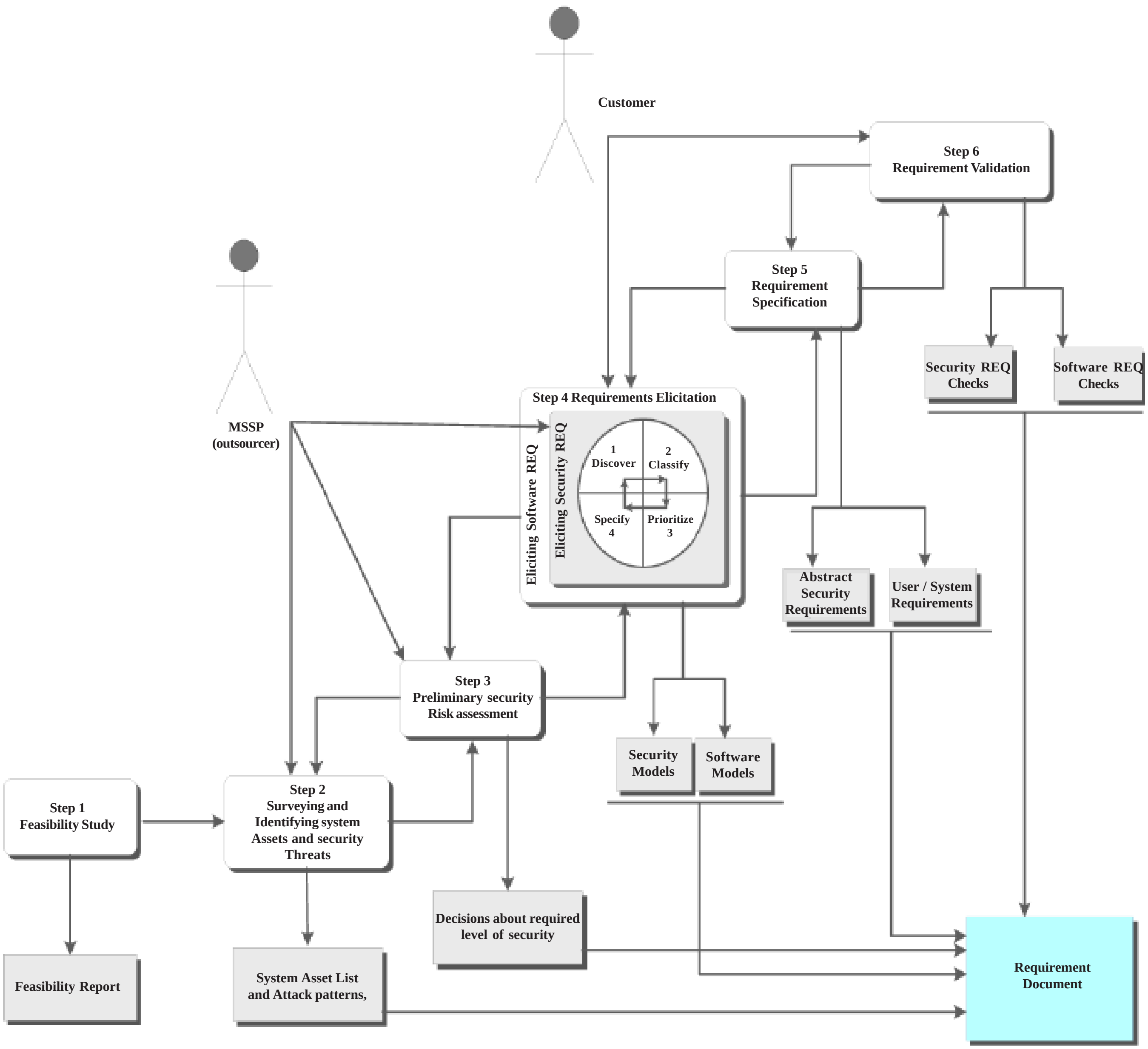


Figure 1. The proposed OSRE Model

2. Related Work

Carnegie Mellon University proposed Security Quality Requirement Engineering Model (SQUARE) which developed to provides a means for eliciting , classifying, categorizing and prioritizing security requirements for IT system [4], [5] and [6]. Chess et, al, in [11] discussed the practices and the activities of building software with security in mind using "*Touchpoints*". Bart De Win, et al in [3] introduced three high –profiles process for the development of secure software , The paper identified the commonalities , discuss the specify of each approach and propose suggestion for improvements. Also, common Web Sites that covered Building security during SDLC in [8], this web site contain links to abroad range of information about sound practices, tools, guidelines, rules, principals and other knowledge to help project managers to build and deploy more secure software system. Several organizations were prefer to gain some security services from an external Outsourcing companies to enjoy its advantages such as cost saving, more expertise and updated technologies etc. in [9] the authors introduced the variables that influence organizational decisions making on global IT Outsourcing. Also Simon Burson in [10] introduced some examples of Outsourcing security considerations and the implications related to each example.

3. Outsourcing Security Requirements Engineering model (OSRE)

In this section we propose a new process model that focuses on considering, defining and specifying security requirements during software requirement engineering process which is the first phase of SDLC. The novelty in our proposed model is adding the Outsourcer represented in MSSP to the development team as anew security member who will be responsible for security consideration during all phases of SDLC, the model focuses only on requirements phase.

Our proposed OSRE model involves six activities (as shown in Figure 1):

- Feasibility study.
- Surveying and identifying system assets and security threats.
- Preliminary security Risk Assessment.
- Requirements Elicitation (Security REQ & Software REQ).
- Requirements Specification.
- Requirements Validation.

3.1 Feasibility Study

Is the first step in which Software engineers and Outsourcer represented in MSSP perform preliminary assessment to the customer's software system in terms of security considerations and software system considerations. Feasibility study should answer to the following questions: (A) Does the customer software system associated with security considerations contribute to the overall objectives of the customer's organizations? (B) Can the customer's system be implemented within schedule and budget using the current technologies and Outsourcing technologies? (C) Can the customer system be integrated with other systems? What are security implications that result from these integrations?. We see that the effective feasibility study should involve six activities:

- **The Customer's Project Scope:** in which software engineers and MSSP define the customer's software system problem and the coordinate security problem and/or opportunity to be addressed. The scope should define the parts of the system affected either directly or indirectly.

- **The Current Analysis:** in which software engineers and MSSP define, understand and analysis the current methods and approaches which used in the development process and the current Outsourcing technologies that used in building security system for the customer.

- **The Customer's project Requirements:** in which software engineers and MSSP should explain how software requirements and security requirements are defined depends on the objectives of the customer project's attention.

- **The Customer's project approach:** in which software engineers and MSSP should identify the recommended solutions to satisfy software requirements and security requirements.

- **The Customer's project Evaluation:** in which software engineers and MSSP should evaluate the recommended approaches (i.e. software approaches and security approaches) in terms of the effectiveness and its practical in the development, also the evaluation involve the cost effectiveness of the selected approaches and solutions and the total customer project cost.

- **The Customer's project review:** in which software engineers and MSSP should assemble all of the preceding elements into a Feasibility Study Report and a formal review is conducted with all parties involved. The review serves two purposes: to substantiate the thoroughness and accuracy of the Feasibility Study, and to make a project decision; either approve it, reject it, or ask that it be revised before making a final decision.

3.2 Surveying and Identifying System Assets and Security Threats

Is a security activity in which MSSP defines, specifies, and classifies the customer system assets which should be protected and identifies security threats and customer software system vulnerabilities which may threaten his software system. Also, In this activity MSSP defines the core security properties which the customer's software system should provide such as (Integrity, confidentiality, accountability, availability and Non-repudiation) to classify the attacks and threats against these properties, so MSSP will be able to identify the customer system vulnerabilities in each asset and the relationship between each vulnerability and specific core security property which should be achieved in the customer's software system. MSSP can perform this activity through developing some of *attack patterns*. each attack pattern describes the vulnerability in each asset and the target core security properties which the attack deliberate to harm and description of the intruders steps to jeopardize the customer system. MSSP can analysis the customer system vulnerabilities and the coordinate attacks which may threaten the customer software system to develop some attack patterns through cycle of four steps (as shown in Figure 2):

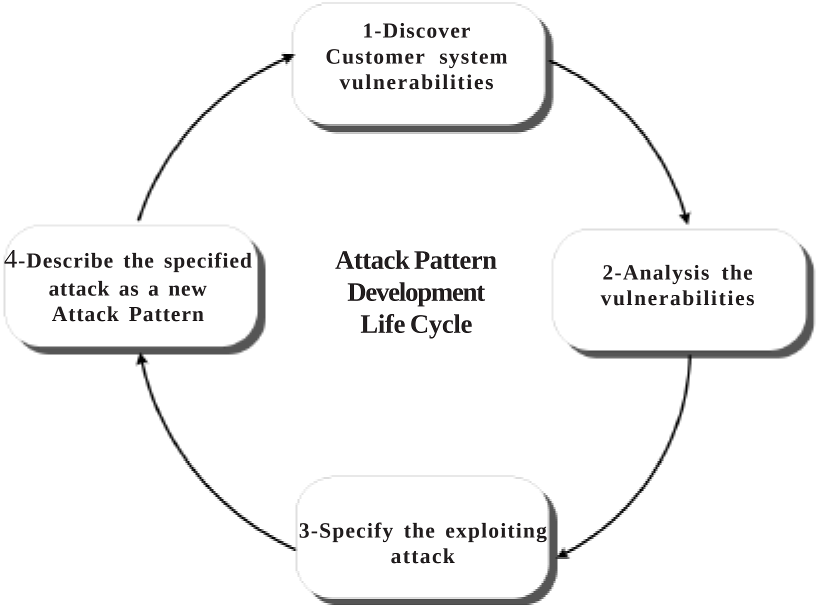

1-Discover Customer system vulnerabilities
2-Analysis the vulnerabilities
3-Specify the exploiting attack
4-Describe the specified attack as a new Attack Pattern
Attack Pattern Development Life Cycle

Figure 2. Attack Pattern Development Life Cycle

The first step focuses on surveying the customer's system in trying from MSSP to discover the expected vulnerabilities in the customer system assets which may be exploited by malicious intruders to penetrate the customer software system. Discovering the vulnerability early will save more effort and cost in the following stages of Software Development Life Cycle (SDLC). The output of this activity will be some of identified security vulnerabilities or weakness in the customer system assets. in the second step, he will analysis these vulnerabilities in detail according to the core security properties (i.e. Integrity, Confidentiality, Non-Repudiation, Accountability and Availability) to stand up the type of vulnerabilities and which asset will be jeopardized of . In third step, MSSP specifies the exploited attack to specific vulnerability in specific asset. the fourth step, describe in detail the specified attack as a new Pattern, each attack Pattern should contain the following attributes *Attack name, Attack prerequisite , Likelihood of the Attack , Risk exposure , Related vulnerabilities and weakness, Method of attack, Attack motivation consequences, Attacker's skills or knowledge required, Attack surface, Solutions and mitigations and Context description*. the new Attack Pattern Development Life Cycle start again until MSSP can specifies all expected vulnerabilities in the customer's system assets.

3.3 Preliminary Security Risk Assessment

Managed Security Service Provider (MSSP) should carry out this step to determine the *Quantitative* and *Qualitive* value of security risks related to the identified vulnerabilities or threats which identified in the previous activity. Assessing security risks which represented in evaluating the identified vulnerabilities will ease eliciting the appropriate security requirements in the next stage of OSRE model. The output of this activity will be some of security decisions which result from evaluating and assessing the customer system vulnerabilities and proposed an appropriate security precaution that should be considered. For assessing and evaluating security risks, MSSP apply two assessment techniques, Quantitative Assessment and Qualitive assessment. *Quantitative Risk Assessment* is a mathematic method which can compute the Security Risk on specific asset depending on two factors, the potential loss (L) that emerge from harming this asset, and the probability (P) of occurring these losses on this asset. So the total Security Risk (R **total**) in all assets can be computed through the following formula:

$$R\ \mathbf{total} = \Sigma\ Li * Pi$$

Using the quantitative assessment approach requires from MSSP to use a *Risk Exposure Matrix technique* to know the level of risk exposure (e.g. Low, Medium or High) . Risk Exposure Matrix output depends on the level of two Factors: potential loss (L), and the probability (P). *Qualitive Risk Assessment* can be described as some of iterative activities (see Figure 3) which Managed Security Service Provider (MSSP) carry out to be has a solid grasp around the identified threats and risks.

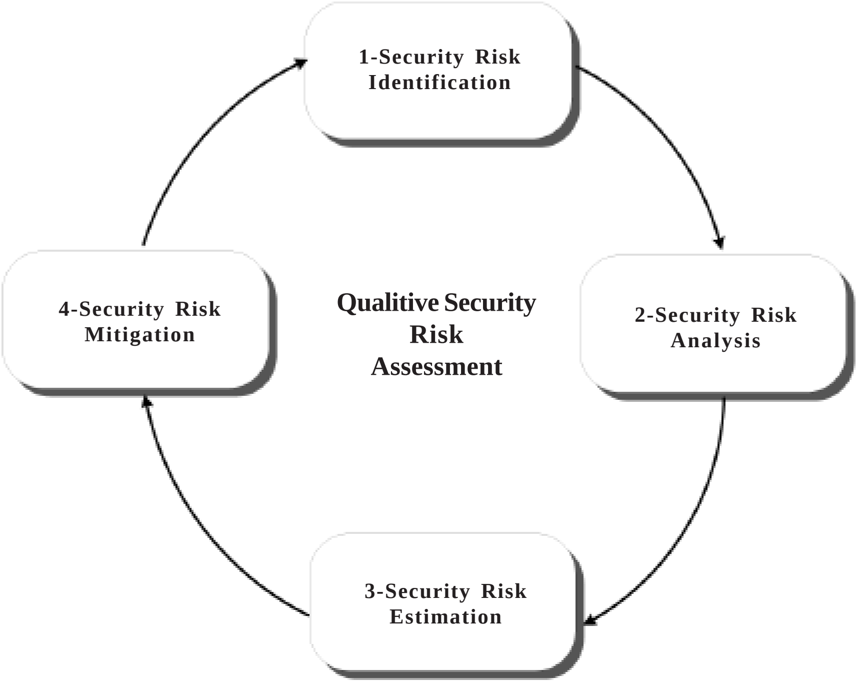

1-Security Risk Identification
2-Security Risk Analysis
3-Security Risk Estimation
4-Security Risk Mitigation
Qualitive Security Risk Assessment

Figure 3. Qualitive Security Risk Assessment methodology

1-Security Risk Identification: concerned with identifying specific attack pattern and identifying its *Risk Path* on specific security Vulnerabilities, security controls which MSSP propose to use, and the technical and business impacts which result from this risk.

2- Security Risk Analysis: in which MSSP analyze each identified *Risk Path* to avoid the technical and business impacts, and define the appropriate preventive measures to reduce the probability of factors that jeopardize the success of security project.

3-Security Risk Estimation: In this activity MSSP can measure the " *Security Risk Exposure*" which all security system of the customer may face. MSSP can use a mathematical methodology to estimate the identified security risks, this methodology called "*Risk Dimension Signature (RDS)*, the output of RDS is a *security Risk Profile* which visualize the *Security* Risk size on the Customer system. through the *Security Risk Profile*, MSSP will be able to identify which dimension need to more attention of security efforts .

4-Security Risk Mitigation: in this activity MSSP attempts to develops a mitigation plans or mitigation strategies to cover , clear or avoids security risks which identified , analyzed and estimated in the previous activities. For Security Risk Mitigation Purpose, MSSP applies seven Steps:

1. Prioritize actions.
2. Evaluate recommended control options.
3. Conduct Cost-Benefit analysis.
4. Select controls.

5. Assign responsibilities.

6. Develop safeguard implementation plan.

7. Implement selected controls.

3.4 Requirements Elicitation (Security REQ & Software REQ)

Is the most major activity in our proposed OSRE model because it involve two requirement elicitations, the general Software Requirements which handled by software engineers(this is not the model focus now), and, Security Requirements which handled and specified by Managed Security Service Provider (MSSP). In this activity the customer will share MSSP to define the appropriate security requirements. The customer should design some of security reports which involve defining and specify precisely the organization policies, security constraints on each asset, number of his employees in the organizations and resource access level of each employee, etc. As we describes in Figure 4 security requirement elicitation can be achieved through iterative four activities:

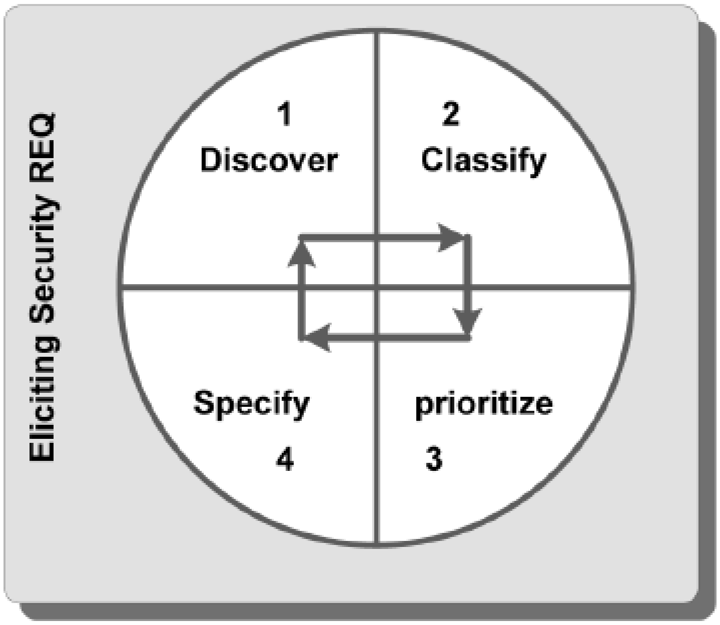


Figure 4. Security Requirements Elicitation Process

1-Discover: is the process in which MSSP gathers security information about the customer system and distill the appropriate security requirements from this information. the appropriate security requirements discovery techniques are: *Abuse Case Diagrams, Misuse Case Diagrams and Attack Trees.*

2-Classify: in which MSSP take the unstructured collection of security requirements, groups related security requirements and organizes them into coherent clusters. The appropriate way which MSSP use is modeling the customer system as groups of assets which require the same type of precautions in an architectural model and associate security requirements to each group of assets.

3-Prioritize: in which MSSP negotiate with customer to resolve the requirements conflicts and identify the priority of each security requirements according to the sensitivity of each asset in the customer system. MSSP can prioritize security requirements using ***Heap Sort Algorithm*** which can be described as following steps:

A- Construct a Binary Tree (BT) for a given array of security requirements.

B- Rearrange BT from the lowest level of BT to the root , by comparing each parent node with its two child nodes from right to left of BT

C- If the child node that associated with security requirement that has larger value (i.e. more important for the customer) than parent node, replace this child node with the parent node.

D- Repeat step 3 till arrive to the state: each requirement associated with the parent node is more important or has larger value than its two child nodes.
E- Remove the root of BT which will associated with the more important security requirement for the customer and put removed root (i.e. security requirement that has largest value or priority for the customer) at the start of a list we called it "*Priority List*" (PL) for security requirements. To refer to the first prioritized requirement that has the largest value or the most critical security requirement.

F- But the most bottom right leaf node of BT in the place of the removed root node as anew root of the Binary Tree (BT).

G- Repeat from step2 again till to remove all nodes of BT and placing each removed one in the *Priority list* (PL) as an ordered list of security requirement according to its importance or its value for the customer.

4- Specify: in which MSSP document each prioritized group of security requirements into a formal documents.

3.5 Requirements Specification

In this activity MSSP produces the general security requirements document which contains all types of security requirements as well as some of outsourcing requirements which the customer system need during software development process. Specifying security requirements depends on the customer asset analysis, threat and risk analysis and the appropriate Outsourced security technology analysis. MSSP can perform security requirement specification through the following stages (as shown in Figure 5):

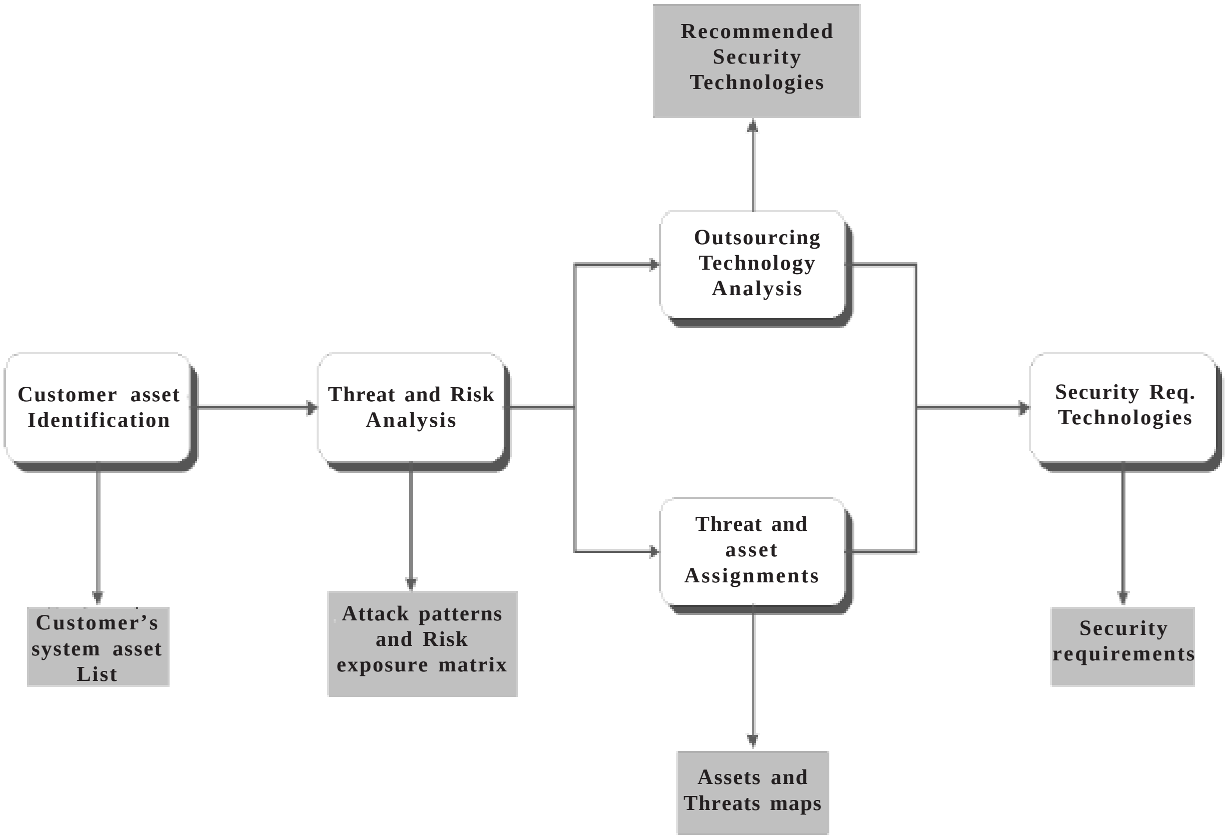

Recommended Security Technologies
Outsourcing Technology Analysis
Customer asset Identification
Threat and Risk Analysis
Security Req. Technologies
Threat and asset Assignments
Customer's system asset List
Attack patterns and Risk exposure matrix
Security requirements
Assets and Threats maps

Figure 5. security requirements specification process

Stage 1: *Identify the customer system* assets in which MSSP will gather all assets in the customer'system and the degree of protection to each one into specific list.

Stage 2: *Threat and risk analysis* in which MSSP identify all developed attack patterns, expected threats and risk exposure matrix which specify the risk level to each attack or threat.

Stage 3: *Threat and asset assignments* in which MSSP develop description maps which describe and explain each asset against the exploiting attack or threat.

Stage 4: *Outsourcing security technology* analysis in which MSSP analysis his available security technologies and propose the appropriate protection security tools against the identified attacks and threats.

Stage 5: *Security requirement specification* in which MSSP write explicitly the document of the final security requirements which the customer system need and the appropriate Outsourcing security services for designing and implementing these requirements.

The output of requirement specification process will contain all User/ System requirements which specified by software engineers and other two types of requirements which specified by MSSP:

- *Actual security requirements* such as Physical requirements, Authentication and Authorization requirements, Access Control requirements, Secure Asset Configuration requirements, Integrity requirements, Non-Repudiation requirements, Monitoring and Auditing requirements and Incident Management requirements.

- *Outsourcing requirements* such as Availability requirements, Customer satisfaction requirements, Outsourcing staff requirements, exit strategy requirements, Outsourced Hardware and Software requirements, disaster recovery and business continuity requirements.

3.6 Requirements Validation

is the last activity in our proposed OSRE model in which MSSP and the customer check the specified security requirements against its realism, consistency and completeness. During security requirement validation process MSSP develops different types of checks on the specified security requirements, these checks include:

1. Validity Checks: the customer may think that security system is needed to perform more security precautions functions, so MSSP perform additional security analysis to identify the required security functions and techniques which the customer's system need.

2. Consistency Checks: by which MSSP and the customer check if there exist conflicts in security requirements. that is, there should not be contradictory security constraints or different description of the same security function.

3. Completeness Checks: by which MSSP and the customer check all specified security requirements to assure that all specified security requirements define all security considerations and security constraints which the customer system asset need.

4. Realism Checks: by which MSSP check his existing security technologies to ensure that all specified security requirements can actually implemented using this technologies.

5. Verifiability Checks: by which MSSP reduce the dispute with the customer using some writing test cases that can demonstrate that all security considerations are met by the specified security requirements.

There are number of security requirements validation techniques which MSSP can use it individually or in conjunction with one another:

1. Outsourced reviews: such MSSP employee some security reviewers who check systematically all security requirements for errors and inconsistencies.

2. Checklist technique: by which MSSP develop some of key security questions to inspect all specified security requirements, the answers of these questions will reflect the security requirements validity checks.

3. Test-case generation: by which MSSP apply some of Test Cases on specified security requirements, if specific test is difficult or impossible to design on specific security requirement, this means that this requirement will be difficult to implement and should be reconsidered again. for instance, MSSP can check the validity of security requirements using Test-case generation technique as shown in Figure 6 through five stages:

1. *Candidate specific security requirement to validate.*

2. *Design security requirement Test-Case.*

3. *Prepare Test-Data.*

4. *Check security requirement with Test- Data.*

5. *Compare results to Test-case.*

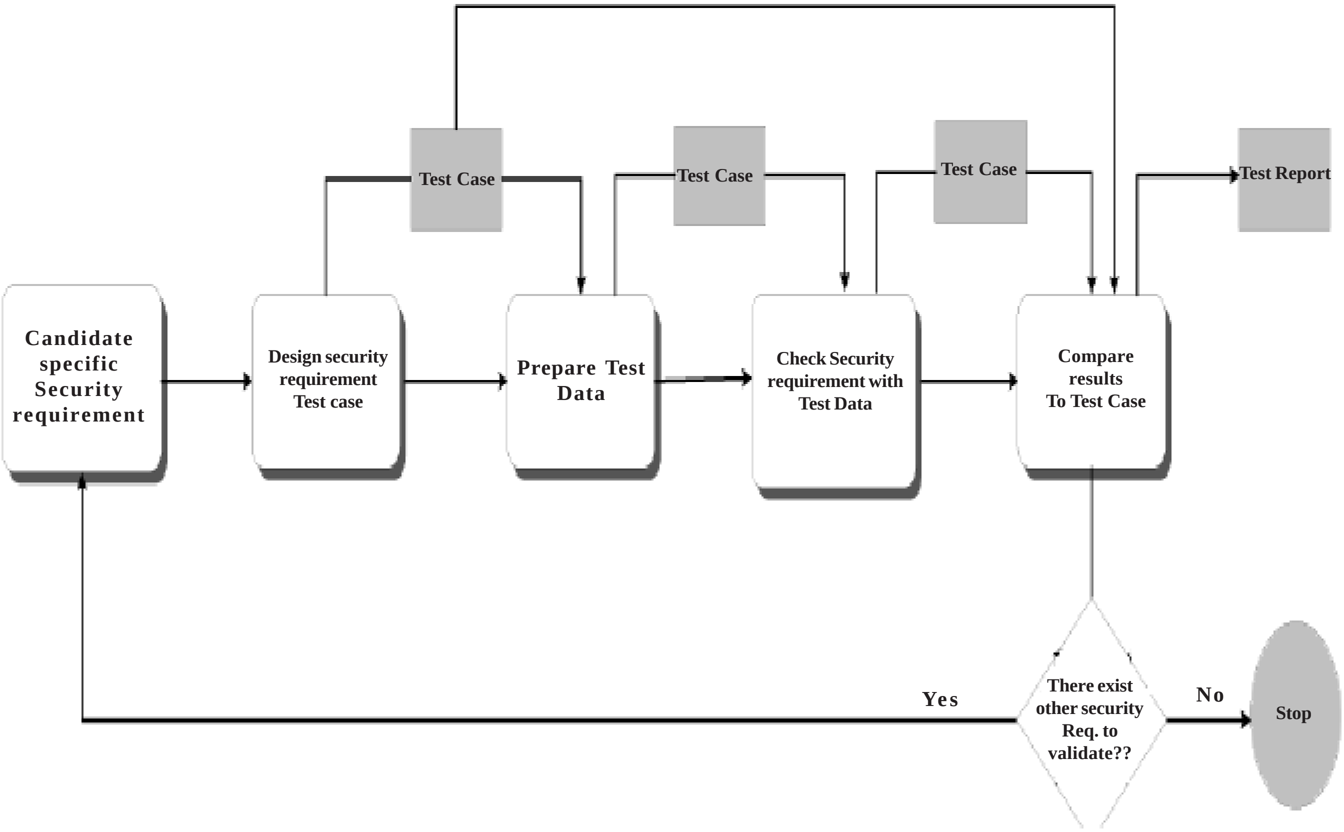

Test Case
Test Case
Test Case
Test Report
Candidate specific Security requirement
Design security requirement Test case
Prepare Test Data
Check Security requirement with Test Data
Compare results To Test Case
Yes
There exist other security Req. to validate??
No
Stop

Figure 6. security Requirements validation using Test- Case generation technique

4. Results

Our results revealed about new process model focused on considering security requirements during the start stage of SDLC (i.e. Requirement phase) using new security person called Outsourcer who represented in Managed Security Service Provider (MSSP). We estimated our new proposed process model (i.e. OSRE model) against Capability Maturity Model Integration (CMMI) to measure its *maturity*, and measured the *Effort* and *Time* which this process model take to complete its six activities using a COCOMO II-Early Design model as the following:

4.1 Measuring OSRE model's Maturity by CMMI

CMMI in software engineering and organizational development is a process improvement approach that provides organizations with the essential elements for effective process improvement. CMMI is the successor of the capability maturity model (CMM) or Software CMM. The CMM was developed from 1987 until 1997. In 2002, CMMI Version 1.1 was released, Version 1.2 followed in August 2006, and CMMI Version 1.3 in November 2010, which address three areas of interest: Product and service development -CMMI for Development(CMMI-DEV), Service establishment, management, and delivery — CMMI for Services (CMMI-SVC), and Product and service acquisition — CMMI for Acquisition (CMMI-ACQ). CMMI exists in two representations: continuous and staged [12]. The continuous representation is designed to allow the user to focus on the specific processes that are considered important for the organization's immediate business objectives, or those to which the organization assigns a high degree of risks. The staged representation is designed to provide a standard sequence of improvements, and can serve as a basis for comparing the maturity of different projects and organizations. The maturity levels in CMMI consist of set of process areas, the maturity levels are measured by the achievement of specific and generic goals that apply to each predefined set of process areas. In our study, we used the continues CMMI to analysis our proposed OSRE model against eight process areas which our proposed model focuses. The continues CMMI for our proposed OSRE model is described in Figure 7 which show that our proposed model has a high maturity level in OPF, PP, RSKM ,REQD and VAL, but has a low maturity level in OT and REQM. Also the model has medium maturity in OPD.

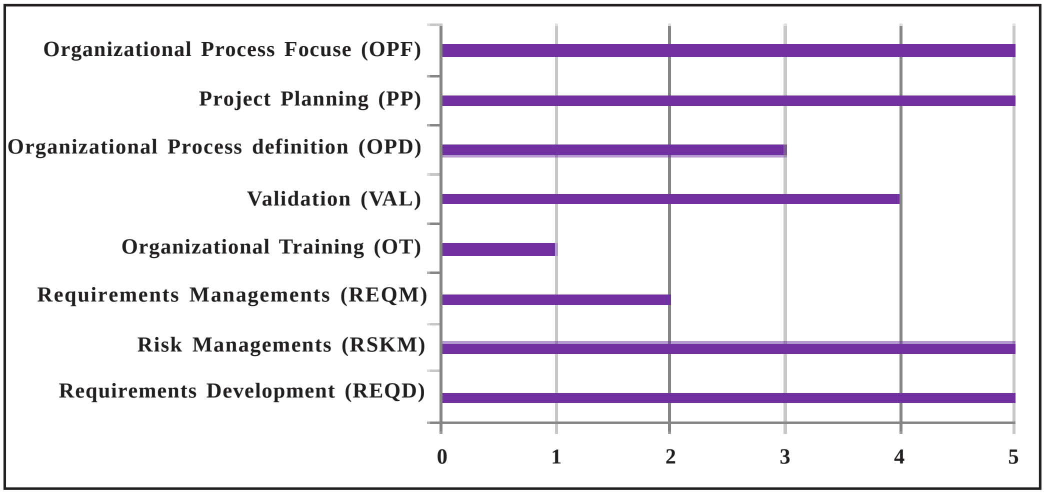
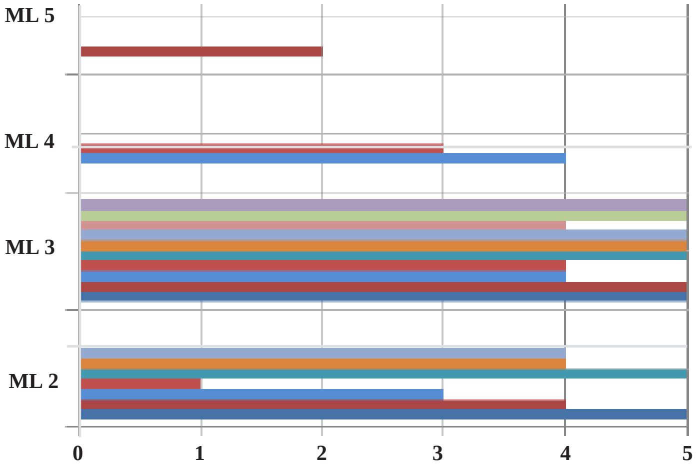


Figure 7. Maturity profile for OSRE based on Continues CMMI

we used a staged CMMI for Services (CMMI-SVC) to assess the Outsourcing organization in providing security services to his customers as described in Figure 8 which show that Outsourcing organization in providing security services have a high maturity in the most of set of process areas in ML2, ML3 and ML4, but fail in ML5.



Maturity level: PAs maturity

ML 2: CM = 5 MA = 4 PRQA = 3 REQM = 1 SD = 5
WMC = 4 WP = 4.

ML 3: CAM = 5 DAR = 5 IRP = 4 IWM = 4 OPF = 5 OT = 5
RSKM = 5 SSD = 4 SST = 5 STSM = 5.

ML 4: OPP = 4 QPM = 3

ML 5: CAR = 0 OPM = 2

Figure 8. Analysis of the developed model based on Staged CMMI

4.2 Effort and Time estimations using COCOMO II-Early Design model

Constructive Cost Model II (COCOMO™ II) is a model that allows software engineers to estimate the cost, effort, and schedule when planning a new software development activity. COCOMO™ II is the latest major extension to the original COCOMO™ (COCOMO™ 81) model published in 1981 [13]. Because our developed OSRE model focused on the requirement phase we chosen *COCOMO II- Early Design model* to estimate the required effort and Time to complete requirement engineering process with considering Outsourcing security requirements elicitation during this stage. As *Early Design model* is an algorithmic cost or effort estimation model we estimated the required effort in range of estimations (Worst, Expected, and Best) to avoid uncertainty in the estimation. The effort estimation using Early Design model could be computed using the following formula:

$$Effort = A\, xsize^{B} \times M$$

Such that ***A*** is a constant factor which depends on local organizational practices and the type of software that is developed. *Size* may be the code size of software or a functionality which expressed in number of *function points or application points*. the value of exponent ***B*** is the complexity of the software development. ***M*** is a multiplier based on seven process attributes which increase or decrease the estimate.

Also, the COCOMOII model includes the following formula to estimate the calendar time required to complete specific project:

$$TDEV = 3 \times (PM)$$

Such that ***TDEV*** is time of development, ***PM*** is the estimated effort and ***B*** is the complexity.

We estimated the required Effort an Time in our proposed model and compared the results with Effort and Time estimations of Requirement engineering process without considering security and Outsourcing in mind (i.e. SOF_REQ_ENG_ WITHOUT_SEC model) and we visualized our *effort and Time* estimations results in Figure 9 and Figure 10 respectively.

5. Discussion

Outsourcing Security Requirements Engineering (OSRE) models focuses on some effective security practices which should be identified during software requirements engineering process. The contribution of this model is that, performing security requirements engineering is not the responsibility of software engineers or some local security specialists in-house but the task will

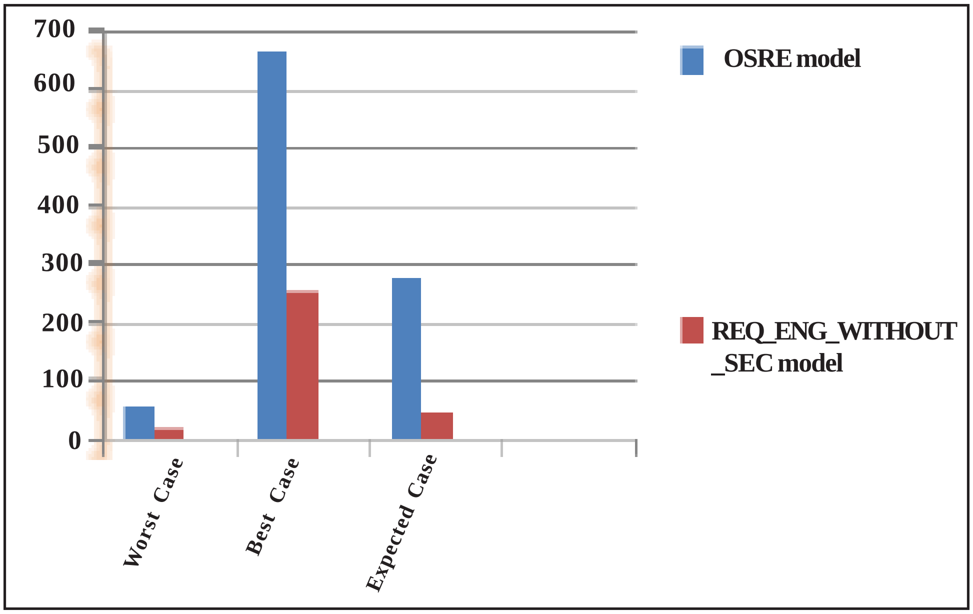


Figure 9. Effort estimation results

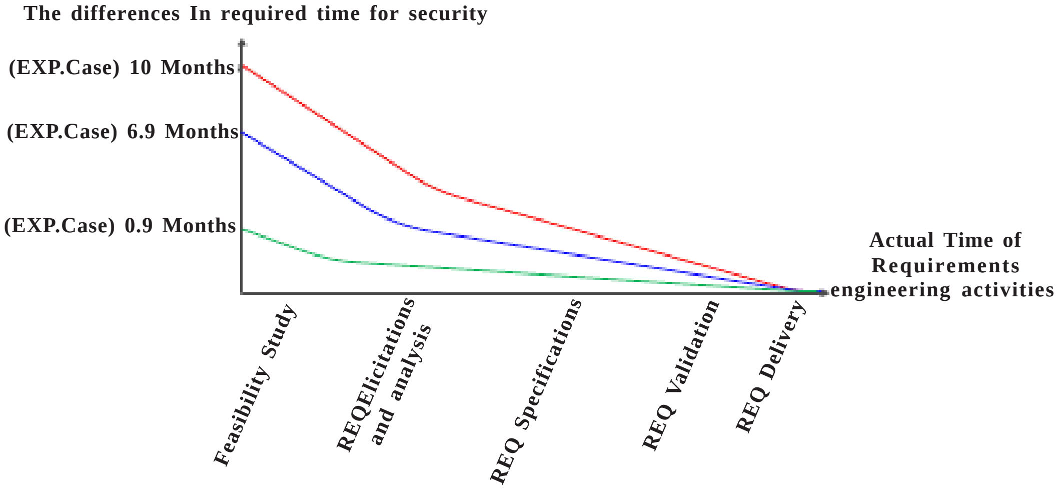


Figure 10. Time estimations for security considerations during software REQ engineering

transfer to an Outsourcing security organization which represented in Managed Security Service Provider (MSSP). after we analyzed our new model based on continues CMMI and evaluated it against some criteria represented in some of process areas which OSRE model focus, the result revealed that (see Figure 7) the model has high maturity in some process areas such as Organizational Process Focus (OPF), Project Planning (PP), Risk Management (RSKM) and Requirements Developments (REQD), but the model has low maturity in Organizational Training (OT) because the model involve MSSP instead of depend on some trained specialists within the customer organization. Also we estimated Outsourcing Organization in providing security services to the customer using the staged (CMMI- SVC), the result revealed that (see Figure 8) Outsourcing organization has high maturity level in most of Process areas of Maturity Level (ML2) which represent the (*Managed level*) and Maturity Level 3 (ML 3) which represent the (*Defined Level*) and Maturity Level 4 (ML4) which represent (*Quantitatively Managed Level*) but Outsourcing Organization need more improvements in the process areas of ML 5 which represent (*Optimizing Level*) . In real, several methodologies explained some security practices which should be applied during SDLC such as Touchpoints [11]. Also some approaches recommended that software engineers should consider security risks during development process such as Operationally Critical Threats, Assets and Vulnerability Evaluation (OCTAVE) [14]. But all these methodologies were overloading software engineers the responsibility of security considerations beside software development process. On the other hand, using our proposed model, the security effort will transferred to a new member of software development team called Managed Security Service Provider MSSP. as we saw in -Figure 9, when we compared our proposed model and the traditional software requirements engineering using COCOMO II- Early Design model, we found that our proposed model require more effort because it has more *function points* will produced from each activity of OSRE model, the additional *Function points* which affected on Effort an Time estimation are several outputs such as, some of developed attack patterns, security decisions about risk assessment, some developed security models, defining and specifying the required security requirements for the customer system and performing some security checks to test the validation of security requirements , the difference in effort denotes to the additional security considerations which should be specifying during requirement engineering, but as our proposed model involves MSSP to perform the additional security effort, this will make software engineers focus more in specifying the basic software requirements and let security requirements work to MSSP. surely, additional security work will require more time considerations , when we analyzed the required Time for security considerations (see Figure 10) we found that customer software requirements engineering require .09 month in *worst case*, 6.9 month in *best case* and 10 months in *our expected case*, these results explains and clears that considering Outsourcing represented in MSSP will save the additional security effort and the additional time for this effort which was overload on software engineers and the owner of the software system.

6. Conclusion and Future Work

Our study revealed about new process model that aims to consider Outsourcing approach which represented in Managed Security Service Provider (MSSP) to be responsible for defining and specifying security requirements during the first stage of SDLC (i.e. Requirements stage).our new process model (i.e. OSRE model) introduced some techniques which MSSP can use to solve security requirements problems such as security requirements discovery, security requirements prioritization, security requirements specifications and security requirements validations. Our results proved that considering security Outsourcer (i.e. MSSP) for specifying security requirements will save more Effort and Time which required for considering security requirements in mind. So considering security from the beginning of SDLC by MSSP will remove heavy overloads on software engineers and the customer and reduces the required Effort and Time for security during the reminder stages of Software Development Life Cycle (SDLC).

Future work will include more partnerships with security Outsourcing organizations for designing and implementing security framework during software design and implementation phase and testing security of software system during software Testing phase till to deploy and configure secure software system for the customers who decided to outsource security considerations to an external security Outsourcing organizations.

References

[1] Julia, H., Sean, A., Robert, J., Gary Mcgraw, E., Nancy, R. M. (2008). Software Security Engineering – a guide for project managers, Addison-Wesley.

[2] Ian, Sommerville. (2009). Software Engineering 9th ed Addison-wesly.

[3] Baret De Win, Ricardo Scandariato, Koen Buyens, Johan Gregoire, Wouter Jooaen. (2009). On the secure software development process: CLASP, SDL and Touchpoints compared, *Science Direct journal information and software technology,* 51, 7 (july).

[4] Gorden, D., Mead, N. R., Stehney, T.,Wattas, N., Yu, E. (2005). System QualityRequirement Engineering (SQUARE); Case study on Asset Management System Phase II. Pittsburg, PA:Software Engineering Institute, Carnego Mellon University .http://www.Sei. CMU.edu/publications/documents/05.reports/05sr0 05. html.

[5] Chung, L., Hough, E., Ojoko-Adams, D. (2005). Security Quality Requirements Engineering (SQUARE) : Case Study Phase III" (CMU/SE- 2006-SR-003). Pittsburg, PA:Software Engineering Institute, Carnego Mellon University. http://www.Sei. CMU.edu/ publications/documents/06.reports/06sr0 03. html.

[6] Chen, P., Dean , M., Ojoko-Adams, D., Osman, H., Lopez, L., Xie, N. (2004). System QualityRequirement Engineering (SQUARE); Case study on Asset Management System Phase II, (CMU/SE-2004-SR-015,ADA43106). Pittsburg, PA:Software Engineering Insti tute, Carnego Mellon University. http:// www.Sei. CMU.edu/publications/documents/04.reports/04sr015. html.

[7] Warren , A . (2004). Outsourcing Information Security, Artech House, Inc.

[8] http://buildingsecurityin.us-cert.gov/

[9] Jianwang, Y. (2011). Determinates of Global IT Outsourcing, www.swdsi.org/swdsi06/proceedings06/papers/mis 03.pdf (Access Aug).

[10] Simon Burson. (2010). Outsourcing Information Security, (January), http://www.techworld.com.au/article/333064/outso urcing-information-security/.

[11] Chess, B., Arkin, B. (2011). Software Security in Practice, *IEEE Security& Privacy,* 9 (2) .

[12] Sally, G. (2008). what is CMMI? NASA Presentation. Accessed 8 Dec.

[13] Boehm, B. W. (1981). Software Engineering Economics. Englewood Cliffs, NJ: Prentice Hall.

[14] Woody , C. Alberts , C. (2007). Considering Operational Security Risk during System Development, *IEEE Security& Privacy,* 5 (1) .