Journal of Theoretical and Applied Information Technology
10th May 2014. Vol. 63 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A REFERENCE MODEL OF SECURITY REQUIREMENTS FOR EARLY IDENTIFICATION AND MEASUREMENT OF SECURITY AWARENESS PROGRAM

Ali Maqousi [1], Tatiana Balikhina [2] and Kenza Meridji[2], Khalid T. Al-Sarayreh[3]
[12] Asstt Prof., Departments of Computer Science and Software Engineering, College of Information Technology Petra University
Asstt Prof., Department of SoftwareEngineering, Hashemite University
E-mail: amaqousi@uop.edu.jo, tbalikhina@uop.edu.jo, kmeridji@uop.edu.jo, khalidt@hu.edu.jo

ABSTRACT

In practice, at software/system requirements assembly stages, the focus is regularly on the software security requirements as usually described at the system level this may lead to explicit security-related product which may be implemented as both in system and software. According to the ECSS standards internal security awareness is restricted to avoid illegal access to the software system and confidential data while the external security requirements related awareness is failing to put off the leak of secure output data awareness and illegal processes. In European, ISO 25021 a amount of terms are afforded to describe many types of aspirant security awareness requirements. This paper accumulates and systematizes these security awareness-related requirements into a standards-based reference model of the software security awareness; In the absence of such a model, such security awareness requirements are definitively assigned at software system testing time, stakeholders find out that a number of Security awareness requirements are neglected and additional efforts should be added to implement such awareness's. Moreover, the proposed model may also be used for identify the functional size of security awareness programs using the ISO 19761 standard. This size may be used for estimation purposes.
Keywords: *Security Requirements, Security Awareness Program, ISO 25021, IEEE-830.*

1. INTRODUCTION

Security awareness's are initially addressed at the early phases of software development as high level functional user requirements or as high level description of software requirements [1-4]. The later on such awareness's should be detailed "As software Functional user Requirements- FUR or in a specific combination of hardware or software".

To discriminate these styles of awareness requirements, functional system offers the required awareness in a system, while nonfunctional awareness's describes how the required awareness functions must perform in a software system. In the software requirements levels, software system awareness's can then be detailed as functional awareness's from user point of view.

In the ECSS standards for the aerospace industry [5-7], many terms are provided to explain various types of security awareness's requirements at the software system level. However, these standards show a discrepancy in their views and coverage of security awareness's.

In the literature, Software metrics techniques are used to quantify of the software systems. Software metrics results are used for set of purposes such as assessing software system quality [8] and complexity [9], for estimating cost and effort [10, 11] as well as for improving software systems process [12].

In spite of the large number of software metrics proposed to practitioners most of them do not fabricate the information required [13] due to a number of weaknesses, including unceremonious definitions [14] and incomplete and/or inaccurate metrics methods [12].

Currently, there subsists no framework for the of functional user software for implementing system security awareness requirements obtained from different views in international standards and in the previous literature. Accordingly, it is difficult to metric these security awareness-related software

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

from functional user point of view, and take them for the cost purposes.

This paper reports on model of software security awareness requirements using international standards, and measures their functionality using the ISO 19761 standard [15].

The paper aims at putting forward an approach for explaining and measuring, software security awareness's using a strategy based neither on our own views nor on individual researchers' view of such type of security, but in international standards of software security awareness's there is no functional requirements from user experiences.

The work purpose is to elect the set of ECSS standards: ECSS [5] is a joint efforts of the European Agency for developing and maintaining standards. This standard addresses the managing product assurance in software projects. This piece of the standard is a "level 2" standard: it is obtained from ISO E12207.

More specifically, the functional methods for cost estimation purposes reached a higher ripeness level. For example, the fundamental terms of the functional measure have been standardized by ISO in [16], while the other five techniques have been aligned by ISO Standards, such as: [15] and NESMA [17]. The ISO 19761 standard identifies the principles, regulations and a procedure for measuring the size of a piece of software systems.

Functional size metrics is used for several purposes: for example to help estimating the developer efforts or help to identify the actual productivity of a finished development Endeavour. Of course there are other reasons to use functional methods too, for more details see [18].

This paper organizes the sections as follows. Section 2 presents the related works. Section 3 presents the Security awareness requirements preparation and analysis. Section 4 describes the identification of standards for security issues. Section 5 presents a standard-based model of requirement for software security awareness's. Section 6 presents measuring the functional security awareness's. Section 7 presents a case study. Section 8 presents the evaluation of the proposed security model. A conclusion is presented in Section 9.

2. RELATED WORK

There are a number of previous researches on security related issues in systems/software engineering projects. For occurrence, in 1993, Chung [19] presented one of the early efforts to detain acquaintance in this field. His work was pursued by that of Mylopoulos [20], who put forwarded viewing for all requirements as software goals, each goal transmitted functional and non functional requirements. Chung [21] followed by Andrew [22] aimed to make system qualities supplementary and functional quantitative in nature, whereas Andrew [22] studied the gaps between the stakeholder images and requirements demonstration. Chung [21] proposed a categorization for the Non functional representing that it is impractical to be expecting from designers to integrate the software body that they cannot willingly identify. While taxonomies aspire for comprehensive of the set of question, these authors suggested in [21] that a two or three-level of categorization could be enough initially, and that there are more than 167 particular types of NFR.

Paech [23] suggested that functional requirements (FR), NFRs, and structural design should be strongly addressed in a rational style, suggesting that NFR can be composable within more sophisticated NFRs and FRs, as well as a design level for judgments.

Moreira.[24] **,** Rosa. [25] **,** Park.[26] , and Glinz [27] have been suggested new techniques for ordering NFRs early on the software systems and development, at the same time as Kaiya [28] proposed a process for stakeholder and their NFRs predilections using a use case diagram of on hand systems.

More recently, Mylopoulos [20] endorsed some Goals Oriented Requirements, and recommended a specific explanation involving the concerns of an Agent Oriented Software.

More recently, Maqousi and Balikhina [29-30] Analyze the behaviour of present users' level of security awareness and also meets your legal, compliance, and audit requirements as well as proposed methods to augmented level of users' awareness to assist users protect their resources such as information, databases, programs, and computer services from any harm or damage.

Kassab [30-31] proposed some models: for separating FRs from NFRs, They also detailed in [32] on an initial r determining of the software size of NFRs based on "Soft-Goal" technique, using the ISO 19761, to deal with the NFR modeling process in early phases in a system.

In corresponding previous work, software engineering designers have been working on the explanation of NFRs, using standards.

In the European work for the aerospace industry [14-19] , security awareness's is defined as a NFR,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

and the research reported now focuses firmly on such wariness's as a NFR.

This paper centers on a one type of software product qualities, that is, security awareness's requirements, and the research approved that there are views for system security awareness's on the basis of international standards, including the use of the ISO 19761 [20] model of software FUR as the guide for the explanation of quantifiable functional requirements.

3. PREPARATION AND ANALYSIS OF SECURITY REQUIREMENTS

ECSS series of standards [5,6 and 7] present software security awareness as a system qualities for software systems: in these standards, the security related requirements are explained as conditions related factors, which might cooperation sensitive information; and the ECSS requires that the system security awareness's should be defined as the requirements baselines [5].

In the ECSS standards, the system security is described as:

- Access control roles for person or group of persons and access control per system or entity.
- Availability for redundant data and automatic restart.
- Data integrity such as integrity with firewall, antivirus, encryption and decryption of data and integrity with different types of system backup (such as automatic, time interval, durability, data versioning and run-time backups).

Software security requirements are also presented in [33] as a non-functional requirement (NFR): this IEEE standard states the aspects that defend the software from unintentional or malicious right to use, amendment, obliteration, or confession. detailed requirements in this area could comprise the necessitate to exploit certain cryptographically methods; to remain explicit log or history data sets; to allocate firm functions to diverse modules; to put a ceiling on communications between several areas of the program and to check data integrity for significant variables.

Furthermore, in [34] lists the security as part of the software functionality to identify the software product superiority.

However, ECSS and IEEE-830 did not put forward a way for measuring security requirements, whilst ISO 9126 presents measures of the result of security organization as a quality of the software product quality, not of the security requirements that have to be built into the software thus not allowing for evaluating the size of such software security requirements: without measurement it is of course demanding of taking such a requirement as a quantitative in an estimation process.

This paper suggests to define security requirements based on international standards, see Table 1.

Table 1. Standards Based Views And Concepts Of Security Requirements

Standards Key Views		Concepts and Vocabulary Describe Security Requirements
ECSS Standards Series 2003-2011	The key views of software security requirements in ECSS standards are described as qualifications, including associated factors, which might cooperation sensitive information. Moreover, the ECSS standards require that the system security shall be defined in the requirements baseline.	European standards proposed the following concepts: • Access control functions for the system, person and groups • Availability for redundant power or data and automatic restart man machined. • System data integrity such as integrity with firewall, antivirus, external PKI.
IEEE-830 Standard 1998	The key views of software security requirements in the (IEEE-Std-830 1998) are factors that defend the software from unintentional or malicious access use.	The IEEE 830 standard uses the following concepts of security requirements: • Cryptographically techniques; • Specific registers or data sets; • Confident functions to many modules; • Confine between some pieces of the program and • Data integrity for significant variables.
ISO 25021 Standards 2002-2004	The key view of software security in ISO 25021 is described as a part of the software functionality to define the software product quality.	ISO 25021 uses the the following concepts and vocabulary for software security: • Access Auditability • Access Controllability • Corruption Data • Encryption Data

Journal of Theoretical and Applied Information Technology
10th May 2014. Vol. 63 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. CLASSIFICATION OF STANDARDS FOR THE REPORTING THE SOFTWARE SECURITY ISSUES

This section engages in the first confront that is, to make out an approach for the software security awareness. The expected result is a standard-based model for software- security awareness requirements. This is realized during the following steps:

Investigation of the set of concepts of security requirements in ECSS, IEEE-830 and ISO 25021 standards to identify the security foundation in system-NFR

The SWEBOK guide was chosen as the foundation for step two while it includes a explaining how to convert the requirements from system to software qualities. For example, the ISO 17959 [33] these behaviors can be used to put together a standard model for software security awareness requirements.

The ISO 19761 is a standard for the functional software, functional size method is hold up by the (COSMIC) and is a distinguished in international standard. The standard ISO 19761 defines the measurement philosophy, some regulations and a complete process for sizing the software systems With COSMIC, the software size must be identified from the beginning. The activating event is an event taking place exterior the boundary of the measured software. The unit of size in this technique is the data movement for the element that moves one or more data characteristics belonging to a single data group such Read (R) or Entry (E), Exit (X), Write (W).

5. A STANDARD MODEL OF REQUIREMENTS FOR SOFTWARE SECURITY

This section proposed a model for the dispersed concepts of security awareness requirements all over standards into a proposed model of security awareness requirements during the defined model proposed in ISO 19761.

The standards model proposed in this section is designed based on the definitions and the understanding of the security awareness requirements as obtained on the definitions of the security in international standards; this is considered as the main reference for the proposed model of software security awareness and ISO 19761 standard for measuring the software size This technique used the following steps to make sure that meets the design of metrology requirements:

5.1 Definitions of Software Security

While the types of security requirements can be divided into three main types:

- Access control roles: only authorized persons can get an access to the data in a system.
- Data availability: redundant data, power and network as well as system automatic restart.
- System data integrity: The data in a system must stay coherent and must not get corrupted by a third party neither by the system itself.

More specification, the security entities to be measured

- Access control functions per person.
- Access control functions per group.
- System authentication for access control functions.
- System data availiability.
- System data integrity and attack detections.

5.2 Building a Standards-Based model for Software Security Requirements.

The Identification of the Entity types in Software Security requirements as follows:

5.2.1 External Security

- **Entity Type 1: Access control roles per person**

Each functional process in access control per person interacts with at least one or more of the security login such as username and password, password change, smart card, single sign-on and automatic logout- see Figure 1.

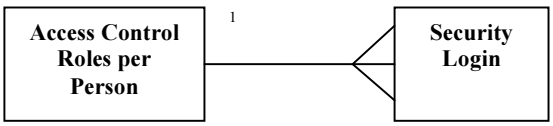

Figure 1: Access Control Per Person

- **Entity Type 2: Access control roles per group of persons or group of systems**

Each functional process in access control per group could share with at least one or more of the security login such as username and password, password change, smart card, single sign-on and automatic logout- see Figure 2.

Journal of Theoretical and Applied Information Technology
10th May 2014. Vol. 63 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

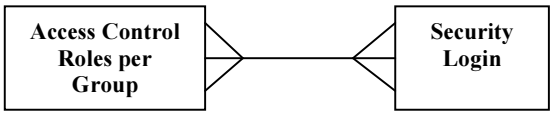

Figure 2: Access Control Per Group

5.2.2 Internal Security

- **Entity Type 3: System data integrity, availability and authentication**

Each functional process in system authentication could share with at least one or more of cryptographic techniques (Encryption and decryption)
Each functional process in system authentication could share with at least one or more of the set of availability data groups such as (Redundant data, power and network).

– One or more of cryptographic techniques can share with system authentication during the security login.
– One or more of cryptographic techniques can share with the set of system data integrity group.One or more of the set of availability data group could share with one or more of set of system data integrity groups- See Figure 3.

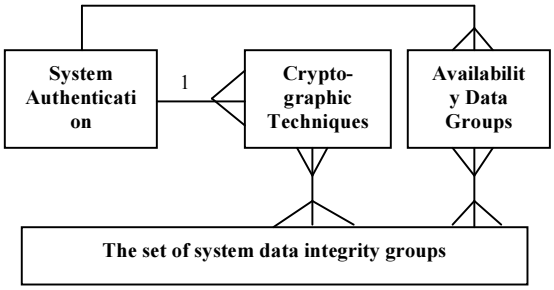

Figure 3: Internal Security

5.3 Model of Relationships for Software Security Requirements.

Classifications of the connections among entity types: In the design of the proposed model of Software security awareness's

- **Entity type 1** measure size of the external security awareness between the access control roles per person and the internal security (represented inside the boundary in figure 4).
- **Entity type 2** measures software size of the external security awareness between the access control roles per group of person or systems and the internal security- see Figure 4.
- **Entity type 3** measure the system authentication links between the cryptographic techniques and the set of available data group, as well as to measure cryptographic techniques and the set of available data group the set of data integrity groups (inside the boundary in figure) - see Figure 4.

6. A PROCEDURE FOR MEASURING THE FUNCTIONAL SIZE OF SOFTWARE SECURITY AWARENESS

Whilst the ISO 19761 illustrates the design method, it does not indicate how to realize it in practice. The implementation of ISO 19761 (ISO-19761 2007) describes the procedure counseled by the ISO 19761 series. Such a measurement contains the following –See Table 2:

Table.2 Measurement Strategy Phase

Measurement Strategy Phase	
Metrics Objective	Measure size of the security awareness's as defined in ECSS- and ECSS-Q-80B and other international standards.
Metrics View	Software perspective point of view.
Results	The size of the software security a wareness's

1. The measurement strategy for the security by defining:
   - The measurement scale
   - The security levels
   - Identifying the security awareness's size and boundary
2. The Mapping stage consists of the elements to set of defined terms aligned with ISO 19761. For the measurement of the security awareness. This stage has been acclimatized as follows- See Table 3:
   - Identify the software security awareness entities and entity types.
   - Define system security groups.
   - Reference ISO model of the Security Requirements.

Table 3 : Security Data Source & Destinations

Source	Objects	Groups
System Authentication	• Per person/group • Per system/entity • Smart card • Biometrics	System data integrity, availability and authentication
Cryptographic Techniques	• Encryptions • Décrépitons	
The Set of Availability	• 24h/day, 7 days/week • Redundant power	

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Data Groups	and network • Redundant data • Automatic restart	
The Set of System Data Integrity Group	• Integrity with firewall • Integrity with anti virus • Integrity with external PKI • Backup types	
Data Destinations		
Access Control Roles per Person	Username & Password • Password change • Smart card • Single sign on • Automatic logout	
Access control roles per group of persons or group of systems		

3. The data movements defined by the proposed model of security awareness's see Table 4.

Table 4: Measurement Security Size For The Proposed Model

Func. Process	Data Description	Movement Type
Access Control Roles per Person		
• ACR per person ENTRY username & password • ACR per person ENTRY to change a password. • ACR per person ENTRY smart card in the system. • ACR per person ENTRY as single sign-on • ACR per person ENTRY to automatic logout		**E** **E** **E** **E** **E**
Access Control Roles per Group of Persons or Group of Systems		
• ACR per group of persons or group of systems ENTRY username & password • ACR per group of persons or group of systems ENTRY to change a password • ACR per group of persons or group of systems ENTRY smart card in the system. • ACR per group of persons or group of systems ENTRY single sign-on • ACR per group of persons or group of systems ENTRY to automatic logout.		**E** **E** **E** **E**
System Authentication		
• Security login SEND all the above entries from access control per person or per group of persons or group of systems to system authentication. • System authentication RECEIVED then READ and WRITES a cryptographic technique during login for the entire exit security login. • System authentication SEND answer to access person or group of person or group of systems if authorized to login to a system • System authentication SEND if authorized to login to a system to check for wanted data in the system availability.		**X (10 times)** **(E, R, W)** **X (10 times)** **E**
Cryptographic Techniques		
• System authentication READ then WRITE the cryptographic technique (encryption) during the login for person or group or system • System authentication READ then WRITE the cryptographic technique (decryption) during the login for person or group or system • Firewall READ then WRITE the cryptographic technique (encryption) for data passing through it. • Firewall READ then WRITE the cryptographic technique (decryption) for data passing through it. • Antivirus READ then WRITE the cryptographic technique (encryption) for data passing through it. • Antivirus READ then WRITE the cryptographic technique (decryption) for data passing through it.		**(R, W) (10 times)** **(R, W) (10 times)** **R & W** **R & W** **R & W**

Journal of Theoretical and Applied Information Technology
10th May 2014. Vol. 63 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

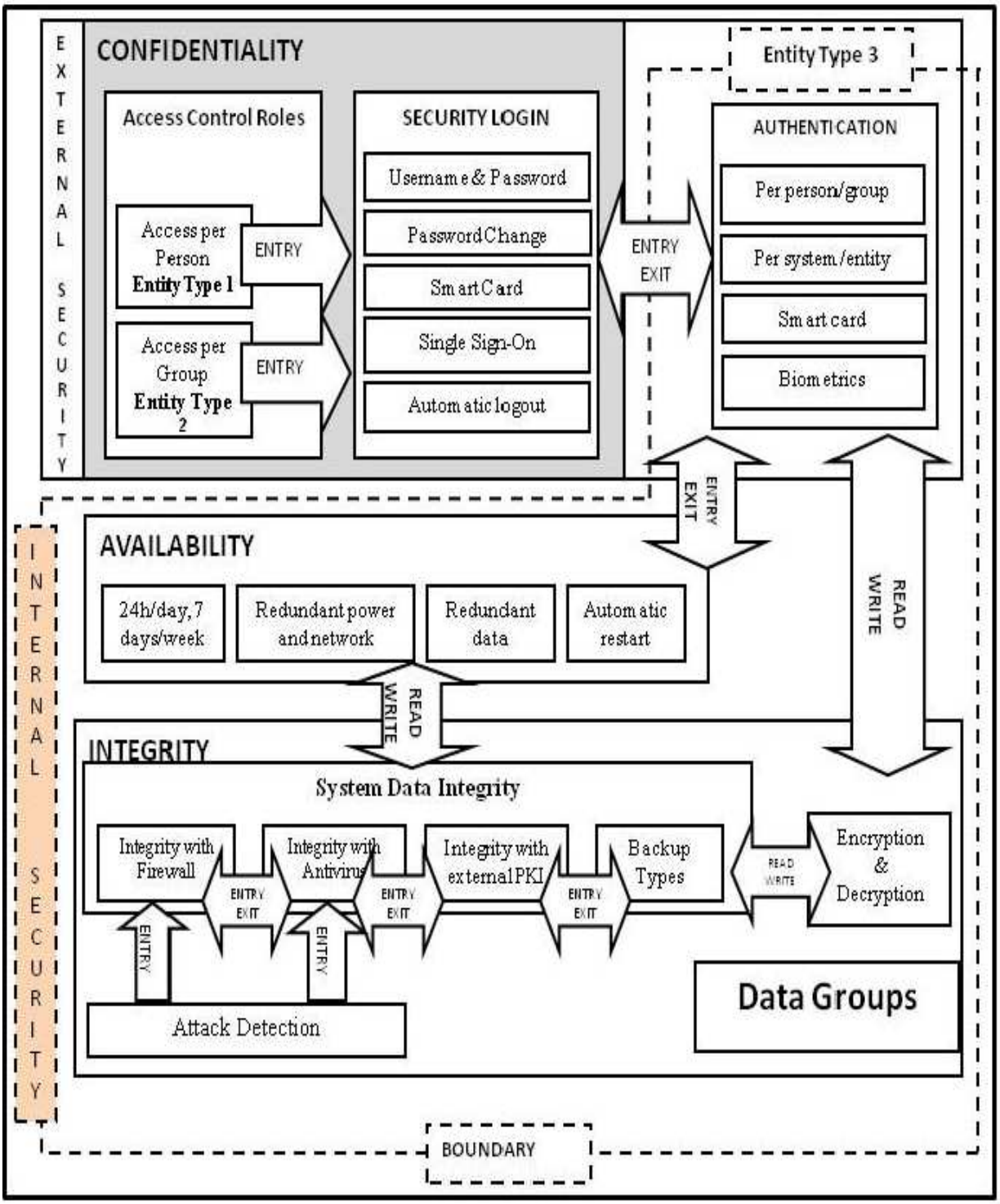

Figure 4. A Standards-Based Reference Model Of Security Requirements

Journal of Theoretical and Applied Information Technology
10th May 2014. Vol. 63 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 4: Measuring the Security Size with respect to A proposed Model (Contd)

Function Processes	Data Movements Description	Data Movement Type
Cryptographic Techniques		
• External PKI READ then WRITE the cryptographic technique (encryption) for data passing through it.		**R & W**
• External PKI READ then WRITE the cryptographic technique (decryption) for data passing through it.		**R & W**
• Backup READ then WRITE the cryptographic technique (encryption) for data passing through it.		**R & W**
• Backup READ then WRITE the cryptographic technique (decryption) for data passing through it.		**R & W**
The set of Availability Data Groups		
• The set availability data groups SEND to authenticated person or group of person to use the available data in the system.		**X**
• The set availability data groups READ data from the set of system data integrity group.		R
• The set availability data groups WRITE data for each part of the set of system data integrity group.		W
The set of system data integrity group		
• Each one of the set system data integrity group RECEIVED data movement from the set of availability data group.		**E**
• Each one of the set system data integrity group SEND data movement to the set of availability data group.		**X**
• Firewall in the set system data group SEND and RECEIVED data movement from antivirus in the same set of the system integrity		**E, X**
• Antivirus in the set system data group SEND and RECEIVED data movement from antivirus in the same set of the system integrity		**E, X**
• External PKI in the set system data group SEND and RECEIVED data movement from antivirus in the same set of the system integrity		**E, X**
• Backup in the set system data group SEND and RECEIVED data movement from antivirus in the same set of the system integrity		**E, X**
• Firewall RECIVED data movement for any attack detection		E
• Antivirus RECIVED data movement for any attack detection		E

In this phase, the basis for these assignment rules is the specific instantiation of the proposed model of security awareness.

With regards to the proposed model, the security awareness size of the internal and external awareness's is defined as follows-See Table 5:

Table 5: Generic COSMIC Measurement Model for Security Requirements.

	The size of the internal security
a.	= ∑ data movement (system data integrity group) + ∑ data movement (Availability of Data Groups) + ∑ data movement (Cryptographic techniques) + ∑ data movement (system authentication).
b.	The size of the external security = ∑ data movement (ACR per person) + ∑ data movement (ACR per group of persons or group of systems)
c.	The size for the Internal and External security = size of the internal security + size of the external security (c = a + b).
d.	The Total Size of the security (Internal and External) $= \sum_{i=1}^{n}$ size of the internal security + $\sum_{i=1}^{n}$ size of the external security n: number of processes for security.

7. PRACTICAL CASE STUDY: "CONFIDENTIALITY AT ACCESS USERS FILES"

The non functional requirements can be defined into different levels, for example there is a NFR in the project level and software product quality level as well as there a NFR in the system level, the proposed a standards-based model in this paper is defined the system-NFR; this system-NFR can be used in the feasibility study if the NFR requirements are defined in the data repository such as ISBSG and Promise data, moreover the proposed model can be also used for a measured project by using ISO 19761 method.

The illustrated example in this section is to measure the system NFR for Security awareness requirements for the confidentiality at access users' files.

This section presents example for using the proposed model of system security awareness's for educational uses for students and trainers.

An organization with five hundred of employees is using online system, where all system's users have access with different permissions to get to a

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

centralized database. In a daily working process each employee uses his/her computer to create files and when he/she finished a complete file he/she will send it to the shared centralized data storage. The original file or part of it in addition to other personal or working files will remain at the user's computer. One of those files might be the credentials of the user. We assume that generated files (data) are classified as sensitive data. In this scenario the organization will be at a risk of attacking the user's computer and the sensitive data will be available to malicious persons. The danger of this incident is falling users' credentials in hands of malicious people who will be able to use them to get access to the system without any notice, forming an undetected intruding to the system. This kind of threats is a most dangers and hard to counterpart by technological solutions. Imagine if it happened what will be the cost of compromising the organization's data or losing it. If we implement a security awareness program that aims at raising users' awareness of all kinds of threats and the way they should act to prevent attacks on their machines we could eliminate or at least reduce the number of such possible threats, and therefore preserve the users and organization's assets.

The measurement of security awareness requirements for the proposed model described in Figure 4.

specifically, The functional requirements of software/system security awareness for a specific instantiation is as:

- **Requirement 1:** Security Awareness program identify using only one of its kind user IDs to allow users to be linked to and held in charge for their accomplishments; the make use of group IDs should be authorized where they are obligatory for business or functioning reasons, and should be permitted and documented;
- **Requirement 2:** inspection that the user has approval from the system holder for the use of the information organism or facility; split approval for access rights from administration may also be suitable;
- **Requirement 3:** Ensuring facility suppliers do not afford access until approval procedures have been completed;

Solution

The specification of security awareness is an explicit instantiation in the proposed model of security awareness's as described in Figure 4.

Table 6 Meseaurement Results Of The Proposed Model

	Data Movement Type
ACR **per Person**	
• ACR per person ENTRY username & password	**E**
• ACR per person ENTRY to change a password.	**E**
ACR **per Group of Persons or Group of Systems**	
• ACR per group of persons or group of systems ENTRY username & password	**E**
• ACR per group of persons or group of systems ENTRY to change a password	**E**
• ACR per group of persons or group of systems ENTRY to automatic logout.	**E**
System Authentication	
• Security login SEND all the above entries from access control per **person** or per **group of persons** or group of systems to system authentication.	**X (2 times)**
• System authentication **RECEIVED** then **READ** and **WRITES** a cryptographic technique during login for the entire exit security login.	**(E, R, W)**
• System authentication SEND answer to access **person** or group of person or **group of systems** if authorized to login to a system	**X (2 times)**
• System authentication **SEND** if authorized to login to a system to check for wanted data in the system availability.	**E**
The Total Functional Size =	**13 CFP**

Table 6 presents software size results of security awareness requirements which described in section 6 and Figure 4 For instance, for access role per person (functional type 1or entity type 1):

- Access control roles per person ENTRY username & password
- Access control roles per person ENTRY to change a password.

This requirement is in contacts to ISO 19761 Entry and Exit, for the size of two ISO 19761 Function Points, or 2 CFP. The total size of this case is equal to 13 data movements for one data group, - see Table 6, bottom line.

8. EVALUATION OF THE PROPOSED SECURITY MODEL

The proposed model of security awareness's in this paper is built based on standards finding software security.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

- The proposed model of the software security is identified of software security awareness as a piece of the application.
- The proposed model in the paper built based on the finding of the ISO 14143-1 and ISO 19761 standards.
- The unit of the measurement in the proposed model of the security is the COSMIC Function Points.
- The tractability model for the proposed model is not described yet.

9. CONCLUSION

Security requirements are normally described at the software system levels, and designers must later allocate these requirements as both software and hardware requirements to be conventional for security awareness's of the system. Surrounded by in the European standards, there are a number of terminologies are described for the security awareness's at the software levels.

This paper proposed a model for software awareness's for the functions needed to deal with the software security.

The contribution of this paper is the proposed model of software security awareness's from user point of view. This model can be taking into consideration as reference model for the classification of software security awareness's.

More specifically, the proposed model of security awareness's in the paper is based on: The set of international standards for the description of the software security awareness's and based on ISO 19761 models of standards.

This model is useful for software designers by providing them whether or not the system engineers are choose the right selection of software NFRs derived from functional user requirements.

The proposed reference model for security requirements presents a technique for measuring these FURs to take these measures into account in software estimation models.

The measurement features the paper is limited to the software security awareness's not for system security awareness's. In the future, It will be interesting to explore whether the proposed measurement can be expanded for all security awareness's requirements.

REFRENCES:

[1] Abran, Alain, Al-Sarayreh, Khalid T. and Juan J. Cuadrado-Gallego, "A Standards-based Reference Framework for System Portability Requirements", accepted for publication in Computer Standards & Interfaces, Elsevier, 2013, DOI:10.1016/j.csi.2012.11.003

[2] Al-Sarayreh, Khalid T., Abran, A. and Cuadrado, J, "A Standards-Based Model of System Maintainability Requirements", Published at Journal of Software Maintenance and Evolution: Research and Practice, John Wiley & Sons, Ltd. 2012, DOI: 10.1002/smr.1553

[3] Al-Sarayreh, Khalid T., Al-Oqily I. and Meridji, K., "A standard-based reference framework for system operations requirements. IJCAT 47(4): 351-363 (2013)

[4] Al-Sarayreh, Khalid T., Al-Oqily I. and Meridji, K.," A Standard Based Reference Framework for System Adaptation and Installation Requirements", In Proceedings of the 6th International Conference on next Generation Mobile , Applications, Services and Technologies (NGMAST 2012), 12-14 Sept 2012, Paris, France, IEEE-CS Press.

[5] ECSS-E-40-Part-1B, " Space Engineering: Software - Part 1 Principles and Requirements", European Cooperation for Space Standaraization,Netherlands, 2003.

[6] ECSS-E-40-Part-2B, "Space Engineeing:Software-part 2 Document Requirements Definitions,." European Cooperation for Space Standaraization,Netherlands 2005.

[7] ECSS-Q-80B, "Space product assurance: Software product assurance,." European Cooperation for Space Standaraization,Netherlands, 2003.

[8] Kan, S. H., "Metrics and Models in Software Quality Engineering", 2nd Edition, Addison-Wesley Professional. Sep 16, 2002.

[9] Podgorelec, V. and M. Heričko,"Estimating Software Complexity from UML Models." ACM SIGSOFT Software Engineering Notes 32(2): 1-5, 2007.

[10] Idri, A., A. Abran, et al., "Evaluating Software Project Effort by Analogy Based on Linguistic Values". 8th IEEE International Software Metrics Symposium. Ottawa, Ontario: 21-30, 2002.

[11] Bourque, P., "Estimating Effort and Cost in Software Projects - ISBSG A Multi-Organizational Project Data Repository for Project Estimation And Benchmarking". Centre for Software Process Technologies. University of Ulster , Northern Ireland, 2003.

[12] Dawson, R. and B. O'Neill, "Simple Metrics for Improving Software Process Performance and Capability: A Case Study " Software Quality Journal 11(3): 243-258, 2003.

[13] Kokol, P. and J. Brest, "Software Complexity Metric with the Critical Vaiue*." IEEE International Conference on Computational Cybernetics and Simulation 1: 494-499, 2007.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[14] Alikacem, E. and H. Sahraoui, "Generic Metric Extraction Framework", 6th INTERNATIONAL WORKSHOP ON SOFTWARE MEASUREMENT and DASMA Metrik Kongress (IWSM/MetriKon). Potsdam , Germany, 2006.

[15] ISO/IEC-19761, "Software Engineering - COSMIC v 3.0A Functional Size Measurement Method. ." Geneva (Switzerland): International Organization for Standardization, 2011.

[16] ISO/IEC-14143-1, " Information technology - Software measurement - Functional size measurement Part 1: Definition of concepts, ISO, ." Geneva (Switzerland): International Organization for Standardization, 2008.

[17] ISO/IEC-24570 (2005). "Software engineering - NESMA Functional Size Measurement Method v.2.2 - Definitions and counting

[18] Forselius, P. (2006). "Faster and more accurate functional size measurement by KISS - keeping it simple "

[19] Chung, L., et al., Nonfunctional Requirements in Software Engineering. Kluwer Academic Publishing, 2000.

[20] John, M. Goal-Oriented Requirements Engineering, Part II. 2006.

[21] Chung, L. and N. Subramanian, System and software architectures. Science of Computer Programming, 2005. 57(1): p. 1-4.

[22] Andrew, J., An Approach to Quantitative Non-Functional Requirements in Software Development. Proceedings of the 34th Annual Government Electronics and Information Association Conference, 2000.

[23] Paech, B., et al., Functional requirements, non-functional requirements and architecture specification cannot be separated -- A position paper. REFSQ, 2002.

[24] Moreira, A., J. Araujo, and I. Brito, Crosscutting Quality Attributes for Requirements Engineering, 14th International Conference on Software Engineering and Knowledge Engineering, Ischia, Italy, 2002: p. 167-174.

[25] Rosa, N.S., P.R. Cunha, and J. F., "G.R.R.: ProcessNFL: A language for Describing Non-Functional Properties". 35th HICSS, IEEE Press, 2002.

[26] Park, D. and S. Kang, Design Phase Analysis of Software Performance Using Aspect-Oriented Programming. 5th Aspect-Oriented Modeling Workshop in Conjunction with UML 2004, Lisbon, Portugal, 2004.

[27] Glinz, M., "Rethinking the Notion of Non-Functional Requirements". 3rd World Congress for Software Quality, Munich, Germany, 2005.

[28] Kaiya, H., K. Osada, and Kayjiri, Identifying Stakeholders and Their Preferences about NFR by Comparing Use Case Diagrams of Several Existing Systems. IEEE Int. Conf. on Requirements Engineering (RE04), 2004: p. 112-121.

[29] Maqousi A. and Balikhina T. "Building Security Awareness Culture to Serve E-Government Initiative", book chapter in Handbook of Research on E-Services in the Public Sector: E-Government Strategies and Advancements, Editors Dr. Abid Al Ajeeli and Yousif Al Bastaki, 2010, Ch 24, Information Science Reference (IGI Global), Hershey-New York, USA. ISBN 978-1-61520-789-3

[30] Maqousi A. and Balikhina T. "User Security Awareness in E-Society", International Arab Conference of e-Technology, IACeT 2008, 5th - 16th October 2008, Amman, Jordan.

[31] Kassab, M., M. Daneva, and O. Ormandjieva., Towards an Early Software Effort Estimation Based on Functional and Non-Functional Requirements. International Conference on Software Process and Product Measurement (MENSURA), Amsterdam, The Netherlands, 2009.

[32] Kassab, M., et al., Non-Functional Requirements Size Measurement Method (NFSM) with COSMIC-FFP, in Software Process and Product Measurement, J.C.-G. Juan, et al., Editors. 2008, Springer-Verlag. p. 168-182.

[33] IEEE-Std-830, "IEEE Recommended Practice for Software Requirements Specifications", 1998.

[34] ISO/IEC-9126, "Software Engineering - Product Quality - Part 1: Quality Model 9126-1", International Organization for Standardization,Geneva (Switzerland), 2004.

[35] ISO-19759, Software Engineering Body of Knowledge (SWEBOK). IEEE Computer Society, 2004.