A Trade-Off Model of Software Requirements for Balancing Between Security and Usability Issues

Article *in* International Review on Computers and Software · January 2016

DOI: 10.15866/irecos.v10i12.8094

3 authors:

Khalid T. Al-Sarayreh
Hashemite University
78 PUBLICATIONS **367** CITATIONS
SEE PROFILE

Lina Hasan
Hashemite University
3 PUBLICATIONS **24** CITATIONS
SEE PROFILE

Khaled Almakadmeh
Hashemite University
23 PUBLICATIONS **66** CITATIONS
SEE PROFILE

A Trade-off Model of Software Requirements for Balancing between Security and Usability Issues

Khalid T. Al-Sarayreh, Lina A. Hasan and Khaled Almakadmeh

Abstract – *The growing dependence on technology drives the pronounced need of developing secure systems that can easily be used by their users. Security and usability have been widely recognized as two non-functional requirements. Different researchers have argued that handling those requirements in a separate manner is considered as a challenge that affects negatively on developing usable systems that can be easily used by their users. Where the security experts most often endeavor to identify and specify the security requirements to resist the potential harms, attacks, and risks that may face the overall system without paying any attention to the resulting issues and difficulties that may face end users when they handle such systems. Over the last two decades, several researchers have introduced different methodologies to facilitate developing usable security systems. Despite that, the research in this area still requires expending more efforts in order to reach a good trade-off between security and usability requirements. For this purpose, this paper aims to analyze the relation between security and usability requirements in order to design a trade-off model that assists in integrating the experiences of both security and usability experts at the requirement stage for reaching a good compromise between both the security and usability requirements.*

Keywords*: Security Requirements, Usability Requirements, Trade-off model*

I. Introduction

Usability is a critical factor that affects the success or failure of the developed systems which operate by the interaction of the end users. In addition, the security factor is not less important in such systems, where it seeks to get the users trust by protecting them from various penetrations and fraud attempts. Security and usability are categorized as two non-functional requirements [1-3]. Both of security and usability requirements should not be added to the system at later stages as add-on features, where many researchers have emphasized the importance of taking such requirements into consideration together and handling them at early development stages [4-6]. However, the development team usually handles these requirements in a separate manner, where they always focus their efforts to develop systems with high-security properties without taking into their account how these properties are easy to use for the end users. Such behavior will lead to develop secure systems that do not satisfy their end users goals [7]. Over the last two decades, several researchers have introduced different methodologies to facilitate developing usable security systems. Despite that, the research in this area still requires expending more efforts in order to reach a good trade-off between security and usability requirements [8-9].

The first step towards developing usable security systems is handling the security and usability requirements together during the requirement phase [10-11]. For this purpose, this research work aims to identify the usability and security requirements in order to analyze the relation between the security and usability requirements for proposing a trade-off model that can be used in the requirement phase. The main goal of the proposed trade-off model is to integrate both the experience of both security and usability experts together in order to handle the security and usability requirements side-by-side and thereby balancing these requirements as much as possible for the aim of developing usable and secure systems.

The rest of the paper is structured as follows. Section 2 identifies the usability and security requirements. Section 3 aims to analyze the relation between the identified usability and security requirements. Section 4 introduces the proposed trade-off model. Section 5 introduces a case study using the proposed model in order to identify usable security requirements for the encryption and decryption functions. Finally, a brief conclusion is pointed out in Section 6.

II. Identifying Security and Usability Requirements in Literature

This section aims to identify the security and usability requirements that will be used later in the trade-off process.

II.1. Identifying Security Requirements

Manuscript received November 2015, revised December 2015

In this paper, the security requirements will be categorized into six main characteristics, which are: *1) confidentiality, it* refers to the ability to prevent unauthorized systems or individuals from accessing the system's data, *2) non-repudiation*, it is about providing evidence to make the parties who participated in a specific event responsible for their actions [12], *3) integrity,* it is concerned with protecting system's data from any unauthorized modification or accidental changes, during the system usage, there are many factors that lead to violating the integrity of system data. Such factors, leading to violation may occur accidently by human errors or hardware failure. In all cases, the system must be able to prevent any violation of data integrity, and also recover data to its original form if any violation or corruption occurs 4) *accountability,* the main goal of accountability is monitoring and tracking the users' actions during using the information system, thereby, those users will be held responsible if they misuse this information [13], *5) Authenticity,* where there are two aspects of authenticity that play a greater role in computing, E-Business, and information security, the first aspect aims to ensure that the data, transactions, and communications are genuine and not corrupted from the original source [14], and the second aspect intends to prove that all the identities' entities (such as users, applications or systems), which are involved in the overall processing, are who they claim to be, and *6) availabilit*y, it refers to the ability to make system's data and resources available for all authorized individuals or systems.

TABLE I
SECURITY REQUIREMENTS

Security Requirements		Description
Confidentiality	Access controllability	It can be achieved by passing through two main stages, the first stage is authenticating the claimed identity of the user attempting to access the system, is correct which is achieved by using different authentication protocols (such as user-ID/password, IC card, and biometrics) and a set of authentication rules The second stage is authorization, it is usually performed after authentication, it intends to determine the privileges of the authenticated user to access system resources and databases [15].
	Data encryption\ decryption	It aims to encrypt system data (either locally stored or in transmission) and that is by transforming it into a non-readable and meaningless form, i.e. ciphered text [16]. This will assist in keeping data safe, where these encrypted data will be re-transformed to its original form) i.e. decrypted (only if it's accessed by authorized users. There are two types of encryption, which are symmetric and asymmetric, in summary, the symmetric encryption aims to encrypt and decrypt the data using the same key. Whereas, in the asymmetric encryption, the sender and receiver should have two keys, one called public and the other private, the message will be encrypted using the sender's private key and the receiver's public key, and decrypted with the receiver's private key.
Non-Repudiation	Digital signature	It aims to introduce a unique, unforgivable and non-reusable electronic signature to be attached into specific data (such as messages) for validating data authenticity and guaranteeing integrity. Moreover, digital signature consists of two main processes, one completed by the signer and the other by the recipient. The first process is creating the digital signature, which can be done using the senders' public and private keys, and the next process is digital signature verification which is only performed by using the sender public key [17]. In addition to the above, a digital signature can also be added by a user to sign a digital document (such as a contract), where several countries handle digital signatures like handwritten signatures.
Authenticity	Authentication protocols and rules	They intend to verify the identity of the communicating partners, and exchange information between them in a secure manner [15]. A variety of authentication protocols has been proposed with different features, such as Secure Sockets Layer (SSL) a protocol for providing secure internet transactions [16].
Accountability	Audit logs	It aims to record the user's access to system resources (such as reading or updating specific records), that will assist in analyzing any violation or misuse that has occurred to any system resources.
Integrity	Hash function	It assists in detecting data corruption by calculating the checksum of specific data and store it in a safe place, if any change occurs on the data, the checksum value will be changed as well. Thereby, the integrity of this data can be verified by re-calculating the checksum value and comparing it with the stored one.
	Backup services	In the case of data corruption, the system must be able to recover it using backup services.
Availability	Network redundancy	It is needed to keep the network always available. The network redundancy uses a replication scheme to store a copy of the live networks current information [18], if the network goes down, the network redundancy will automatically kick in to maintain network availability.
	automatic restart service	Various failures may occur in the system process causing the system to crash and preventing the user from restarting it. An automatic restart function will help force the corrupted processes to restart
	Automatic backup	if any data corruption happened, the data will be automatically recovered to be available

II.2. Identifying Usability Requirements

In this paper, the usability requirements will be categorized into two main parts, the first part can be evaluated during the system development stages, where it consists of six main characteristics, which are: *1) learnability*, it evaluates the degree to which specific users are easily able to learn the system and use it, *2) appropriateness recognizability*, it reflects the users' ability to choose the system or product which provides

the appropriate functionalities for their intended objectives, *3) operability*, it aims to evaluate to which extent the users are capable of easily operating and controlling system attributes, *4) user error protection*, it reflects the system's ability to protect its users from making errors, *5) user interface aesthetics*, it assesses the users' satisfaction and pleasure with regard to system user interfaces aesthetics, and *6) universality*, it reflects the system's ability to accommodate a variety of users who have different cultural backgrounds. Each one of these characteristics is decomposed into a set of sub-characteristics. Table 2 summarizes the decompositions of each characteristic.

TABLE 2
USABILITY REQUIREMENTS DURING THE SYSTEM DEVELOPMENT STAGES

Usability requirements		Description
Learnability	Completeness of user documentation	It aims to make sure that all systems functionalities or any help facilities are described completely and correctly [19].
	Memorability	It aims to ensure that the casual users, who have not used the system for a period of time, are able to remember the steps of using particular functions without the need to re-learn them from scratch [20].
	Time to learn	It aims to ensure that the users are able to learn and use specific functions through a specific period of time [21].
	Ease of help access	It aims to ensure that the system provides help items of the system's functions that can be easily accessed by the users [22].
Appropriateness recognizability	Description completeness	It aims to ensure that all the functions which are provided by the system are described correctly and completely in the system description [22].
	Understandable I/O	It refers to the user's ability to easily understand all the required data inputs and the provided outputs in a short period of time [22].
Operability	Operational consistency	It refers to the necessity of uniformity in the appearance and behavior of system's components (such as functions or messages) that have similar properties, this assists in making a system easy to learn and remember [19].
	Understandable system messages	Understandable system messages (such as error, warning, confirmations and notifications messages) assist users to clearly understand the content of these messages and also to memorize the important ones.
User error protection	avoidance of incorrect operation	The system's functions must be implemented with specific code or design that assists in preventing the system from implementing any incorrect operation that may produce serious defects.
	User error recoverability	In case the system's users have caused the occurrence of errors, the system must be able to recover and correct such errors easily [22].
User interface aesthetics	Appearance customizability of user interface	It refers to the users' ability to customize elements in the user interface. It can encourage and attract the users to operate the system for long periods of time, thereby aiding in improving the system's learnability and Memorability [22].
Universality	Accessibility of physical disability	It aims to develop the system with tools and specific features to make it usable by a wide possible range of users regardless of the disabilities that they may have [23]. For example, ensuring that the text size in the developed system can be adjusted to be accessible for those who have poor visibility.
	Cultural universality	It refers to the ability in using the system by those people who have different culture background [24].

Whereas the second part of usability requirements aims to be evaluated while using the system in realistic context, it decomposes into four main characteristics, which are: *1) effectiveness*, it reflects the capability of specified users to use the system in a particular context and achieve their goals in a complete and accurate manner, *2) efficiency*, it evaluates to which extent a specific user consumes resources in order to achieve their goals effectively, *3) productivity*, it reflects the achieved effectiveness level with regards to the consumed system resources, by users and the system, and *4) satisfaction*, it reflects the users' satisfaction level to use the system in a particular context.

Each one of these characteristics is decomposed into a set of sub-characteristics. Table 3 summarizes the decompositions of each characteristic.

TABLE 3

USABILITY REQUIREMENTS WHILE USING THE SYSTEM IN REALISTIC CONTEXT

Usability requirements		Description
Effectiveness	Task completion	It aims to objectively measure the ratio of the system's tasks that are completely executed to achieve the user's goal [22].
	Task effectiveness	It refers to the ratio of the system's tasks that are executed completely in a correct manner [22].
	Error frequency	It intends to objectively measure the errors committed by a user occurring during user's working on the system's tasks [22].
Efficiency	Task duration	It determines the time spent by a user to complete a specific task [25].
	Task efficiency	It aims to measure the ratio of achieved task objectives per unit of time [22].
	Relative user efficiency	It determines how efficient a regular user achieves a task in comparing the efficiency of an expert to achieve the same task [22].
Productivity	Economic productivity	It relates task effectiveness to cost, where the cost can be a user's time or the time of others giving assistance to complete the task [23].
	Relative number of user actions	It aims to ensure whether the user performs the minimum number of actions needed [22].
Satisfaction	Usefulness	It aims to assess the extent that users are satisfied with their achievement of goals, it can be evaluated by evaluating the satisfaction scale and number of users' complaints. The satisfaction scale can be measured by questioners to determine the extent of user satisfaction. Whereas the number of users' complaints reflects the ratio of customers who make complaints against the system [21].
	Trust	It aims to ensure that the users have confidence that the behavior of the system will be as intended [23].
	Pleasure	It aims to ensure that the users get pleasure from achieving their personal needs.
	Comfort	It aims to ensure that the users are physically comfortable when using the system.

III. Analyzing of The Security and Usability Relationships

Based on the previous section, this section aims to analyze the relation between security and usability requirements, where the main goal of this section is to clarify the usability requirements that are required to be achieved through each security requirement in order to identify such requirement in a usable manner. However, for more clarity, Appendix 1 summarizes the relation between security and usability requirements.

III.1. Usability of Authenticity

As previously clarified, there are two main aspects of authenticity, the first one aims to authenticate all the identities' entities (such as users, applications or systems), in order to prove that they are who they claim to be. However, in order to allow users to authenticate themselves in a usable manner, the following usability requirements must be taken into account:

1) *Appropriateness recognizability:* the users must be able to understand the required data inputs to authenticate them, and the data output that results from the authentication process. In addition, the authentication process must be described in the system description.
2) *Learnability:* the users must easily learn and remember the authentication steps, in addition to easily remembering the entities that are used to authenticate them (such as ID & password). In case the users face a problem during authenticating themselves, the system must provide online help to allow them to handle such problems easily. Moreover, the steps of authenticating users must be documented in details in the user documentation
3) *Operability:* the system must provide understandable and clear messages to allow users to easily understand its content.
4) *User error protection:* the system must be able to prevent users from making errors during authenticating themselves. For example, prevent a user from leaving the ID or password field empty by introducing an understandable error message for them.
5) *Universality:* it will be preferable to provide more than one authentication method and allow users to choose the suitable one (such as text-based password and One-Time-Passwords), that will assist the users who have different disabilities to authenticate themselves easily.
6) *User interface aesthetics:* it will be preferable to allow users to customize the user interface elements of the authentication page, such as resize the font and change the background theme.
7) *Effectiveness:* the users must be able to authenticate themselves completely, correctly and with fewer errors.
8) *Efficiency:* the user must be able to achieve the objective of the authentication task through specific time.
9) *Satisfaction:* the users must feel satisfied and comfortable while authenticating themselves.
10) *Productivity:* the users must be able to authenticate themselves in an effective manner with consuming less resources.

The second aspect of authenticity intends to ensure that the data, transactions, and communications are genuine and not corrupted from the original source [14],

usually, that is achieved automatically without requiring the end users to perform any actions. However, in this case, the users must get a notification message as feedback about the authenticity status. In addition, if something wrong occurred, the users should receive a clear error message clarifying when and why the error happened, and how to handle it and also provide online help to allow the users to handle such errors easily. Moreover, the utilization of such authenticity aspect must be also described in the system documentation.

III.2. Usability of Confidentiality

Confidentiality is decomposed into data encryption/ decryption and access controllability. The process of data encryption/ decryption can be accomplished automatically without requiring the end users to perform any actions (such as the communication channel encryption protocol), or it can be accomplished where end users encrypt/decrypt specific data (such as a message) using public and private keys. However, when designing encryption and decryption mechanisms that require performing actions by the end users, the following usability characteristics must be taken into account:

1) *Appropriate recognizability:* the users must be able to understand the required data inputs to encrypt and decrypt data, and the data output that results from the encryption and decryption process. In addition, the encryption and decryption process must be described in the system description.
2) *Learnability:* the users must be able to understand the data inputs in the encryption or decryption processes, as well as the data outputs that will result from these processes. They also must be able to easily learn and memorize the steps of encrypting and decrypting specific data, even if they have not used the system for a period of time. In case the users face a problem during the data encryption/ decryption processes, they must be able to get online help to facilitate in handling such problems. In addition, the steps of encrypting and decrypting data must be clearly documented in the user documentation.
3) *Operability:* both the encryption and decryption processes must be accomplished in a consistent manner. In addition, the users must get understandable errors, notifications, warning messages to allow them to handle such messages effectively.
4) *User error protection:* the system must be able to prevent users from errors during the encryption process. For example, prevent a user from completing the encryption process if wrong password is entered.
5) *Universality:* it will be preferable to support the encryption and decryption process for different languages.
6) *User interface aesthetics: it* will be preferable to allow users to customize the user interface elements of encryption and decryption pages, such as resize the font and change the background theme.
7) *Effectiveness:* the user must be able to complete the encryption and decryption tasks completely, and correctly with fewer errors.
8) *Efficiency:* the user must be able to achieve the objective of the encryption and decryption tasks in an effective manner through a specific time.
9) *Satisfaction:* the users must feel satisfied, and comfortable while performing the encryption and decryption tasks.
10) *Productivity:* the users must be able to complete the encryption and decryption tasks in an effective manner with consuming less resources.

Furthermore, when designing usable data encryption/decryption mechanisms that automatically encrypt/decrypt data without requiring users to perform any actions, the users must get a notification message as feedback about the encryption or decryption status. Moreover, if something wrong occurred, the users should receive a clear error message clarifying when and why the error happened, and how to handle it and also provide online help to allow the users to handle such errors easily.

Access controllability is achieved by authenticating and authorizing users who attempt to access the system. Security experts achieve the goal of access controllability by selecting the appropriate authentication protocols and putting suitable rules. Moreover, when authorizing the system's users, the following usability characteristics must be applied:

1) *User error protection:* the user must be prevented from producing errors by accessing or altering specific data. In case that a user has made a mistake by manipulating specific data, he/she should be able to recover it.
2) *Operability:* it is important to introduce understandable messages to allow users to clearly understand the situation. For example, introduce error messages for unauthorized users to clarify why they are not able to access or alter specific data.
3) *Learnability and appropriate recognizability:* the user documentation and the system description must clarify the permeations of each user type.

III.3. Usability of Non-Repudiation

Digital signature is a commonly used mechanism for ensuring non-repudiation and authenticity, where it can be added by a user to electronic documents or attached to data. However, when designing usable digital signature schemes which require users to electronically sign electronic documents, the following usability

characteristics must be taken into consideration:

1) *Appropriateness recognizability:* the users must be able to understand the required data inputs to sign a document, and the data output that results from verifying a signature. In addition, the signing and the verifying process must be described in the system description.
2) *Learnability:* the users must be able to learn and memorize the signing and verifying steps easily even if they have not used the system for a period of time. In addition, the users must get understandable messages that keep them vigilant for any event. For example, if any error occurred during the signature signing or verifying process, understandable error messages should be provided in order to allow the users to know why, when and how the error has occurred. Another example, it's important to introduce a clear notification message to inform users who signed the signature, the date of signing and if it has been verified or not. This step helps in increasing the users' trust and confidence in adopting such signatures. Moreover, the steps of signing and verifying signatures must be clearly documented in the user documentation.
3) *User interface aesthetics:* the signature layout in the user interface must be resizable in order to allow users to sign the document in a comfortable manner.
4) *User error protection:* the users must be prevented from making mistakes, such as modifying the document after signing it.
5) *Operability:* both the signing and verifying processes must be accomplished in a consistent manner.
6) *Effectiveness:* the users must be able to perform the signing task and signature verification task completely and correctly with fewer errors.
7) *Efficiency:* the users must be able to perform the sign task and signature verification task in an effective manner through a specific time.
8) *Productivity:* the users must be able to complete the sign task and signature verification task in an effective manner with consuming less resources.
9) *Satisfaction:* the users must feel satisfied and comfortable while using the digital signature feature.

Furthermore, when designing digital signature scheme that is achieved by using public and private keys, the developers should take into consideration the same usability characteristics that should be achieved when designing the encryption and decryption scheme that requires users to encrypt data using public and private keys.

III.4. Usability of Accountability

Accountability can be achieved by audit logs, such logs aim to record the user's access to the system's resources. Hence, if any violation or misuse occur in system's resources, the audit logs will be analyzed to detect the responsible parties that caused such violation or misuse. Many systems introduce the audit logs as a feature for their users to monitor their profiles. However, in order to design usable audit logs, the following usability requirements must be taken into consideration:

1) *Learnability:* the steps of opening, reading and analyzing the audit logs must be clearly explained in the user documentation. Furthermore, the users must be able to easily learn and memorize the steps of opening, reading and analyzing the audit logs even if they have not used the software for a period of time. In addition, the system must provide clear system messages for their users. For example, if the system administrator or users want to delete the audit log, the system must introduce a warning message that clarifies the effect of such action. In case system administrators or users face a problem dealing with the audit logs, the system must provide online help in order to assist in solving such problems.
2) *Appropriateness recognizability:* the system administrators or users must be able to understand the output data recorded in the logs. In addition, the utilization of audit logs must be described in the system description.
3) *Universality:* it will be preferable to support the appearance of audit logs with different languages.
4) *User error protection:* avoid system administrators or users to alter the content of audit logs. In the case of system administrators or users cause an error to the audit logs, the system must be able to recover such error. For example, the administrator has deleted specific audit log accidently, the system will recover such error by backing up the deleted log.
5) *Operability:* the system must provide understandable error messages to allow the users to handle them effectively.
6) *User interface aesthetics:* allow the user to resize the font size and type of their audits log.
7) *Effectiveness:* the users must be able to open and read their audits log completely and correctly with fewer errors.
8) *Satisfaction:* the users must feel satisfied and comfortable while opening and reading the logs.

Usually, the duration of keeping the system audit logs is stated in the requirement specification. However, the utilization of audit logs and their keeping time duration should be mentioned in the system description document. In addition, if the keeping duration for specific audit log is on the verge of completion, it's important to provide notification messages for the concerned authorities to inform them that the audit log will be renewed after a specific time. In addition, in case the audit logs have

been renewed, the concerned authorities must get a notification message to inform them about the audit log's status.

III.5. Usability of Data Integrity

Data corruption prevention plays an important role in achieving the goal of data integrity, it decomposes into a hash function and data backup. However, regarding the utilization of the hash function, the system must automatically monitor the checksum value of the system's data, in case any corruption should occur, the checksum value will be changed. In this situation, the system should inform the users by a notification message to clarify how and when the corruption happened, it in return will assist the users to immediately handle such corruption by recovering the corrupted data.

Furthermore, the following usability characteristics are necessary for allowing users to easily conduct a backup for the corrupted data:

1) *Learnability:* the steps of recovering data must be documented in the user documentation. Moreover, the system's users must be able to learn the recovering process in a short time and memorize it easily. In case the users face a problem during the backup process, the system must provide online help to facilitate in handling such problems.
2) *Operability:* the system must provide understandable error messages to allow the users to handle them effectively, and introduce notification messages to provide the users feedback about the recoverability status.
3) *Appropriateness recognizability:* the users must be able to understand the backup files as output and input. In addition, the utilization of the backup function must be described in the system description.
4) *Universality:* it will be preferable to support the backup process with different languages.
5) *User error protection:* avoid letting users manipulate sensitive system backup files.
6) *User interface aesthetics:* it will be preferable to allow users to customize the user interface elements of encryption and decryption pages, such as resize the font and change the background theme.
7) *Effectiveness:* the user must be able to complete the backup tasks completely, and correctly with fewer errors.
8) *Satisfaction:* the users must feel satisfied, and comfortable while performing the backup tasks.

III.6. Usability of Availability

As previously mentioned in Section 2, availability is decomposed into network redundancy, automatic restart function, and backup. All of these decompositions are automatically used by the system to maintain the system availability, in case that the system becomes unavailable, when the system becomes available, the users must get a notification message as an apology about the system unavailability. In addition, the utilization of network redundancy, automatic restart function, and backup should be mentioned in the system documentation.

IV. The Proposed Trade-off Model for Balancing Security and Usability Requirements

Security and usability are categorized as two non-functional requirements, many researchers have emphasized the importance of handling such requirements at much early development phases [25]. However, it's well recognized that requirement phase plays a critical role in the success of the developed system, where the success of all the next development phases is primarily depending on completing this phase in an efficient and effective manner. For this purpose, this section proposes a trade-off model that aims to be used during the requirement phase, the main goal of this model is to integrate the experiences of both the security and usability experts in order to identify and analyze the security and usability requirements side by side for developing systems that provide usable and secure features.

Fig. 1 illustrates the proposed trade-off model, where during the requirement specification, the security experts will specify the security requirements by describing how these requirements will be carried out with respect to the system constraints (such as environment and cost constraints). It's an important to make the security experts work in parallel with the system analysts, where the security specification is represented as a part of the whole system requirement specification. Next, the security experts should conduct a risk assessment for identifying the risks that may result from such requirements, their likelihood and impact and then identify the required actions to control it. Risk assessment process can be conducted through several development stages with different levels of abstraction [26]. However, after conducting a risk assessment, the security requirements specification may be updated to improve them.

Thereafter, the security and usability experts should communicate and coordinate with each other in order to assist the usability experts to clearly understand the security requirements specifications which are related directly or indirectly with the end users. The usability experts in return should analyze each of these requirements and identify the usability requirements that should be achieved to express the security requirements in a usable manner (see Appendix 1).

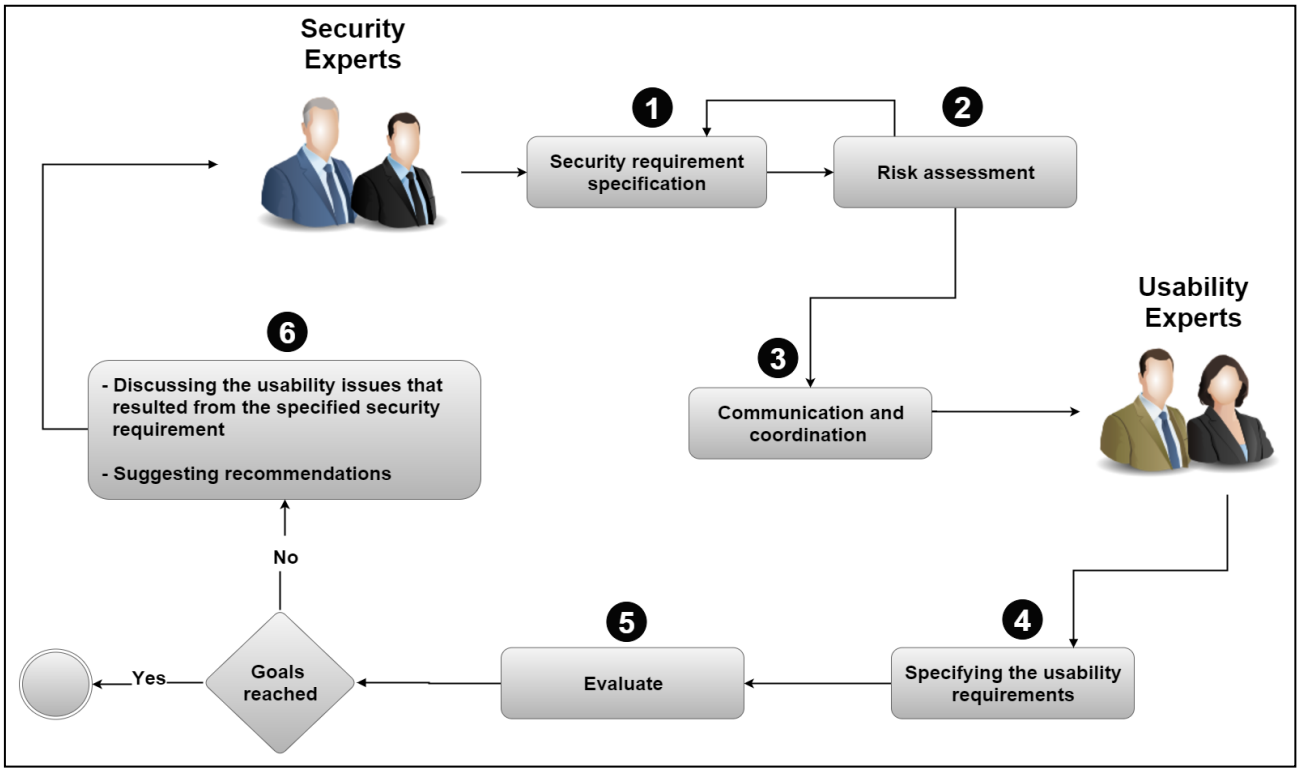

Fig. 1.The proposed security and usability requirements trade-off mode

In fact, identifying the usability requirements that are related to each security requirement will not be enough, where the usability experts should evaluate these usability requirements in order to ensure that they can be achieved through the specified security requirement, the evaluation process primarily depends on the user experts experience, where a cognitive walkthrough evaluation can be used in this step, it aims to evaluate usability by an expert or a group of usability experts, they will identify specific usability requirements and walk through the described security requirements to ensure that all the identified usability requirements can be achieved through the specified security requirements. It is important to remark that at this phase, the usability experts can evaluate only the usability requirements that are clarified in the first usability model, which are: appropriate recognizability, learnability, user error protection, operability, user interface aesthetics and universality. Whereas the requirements of the second usability model, which are: effectiveness, efficiency, productivity, and satisfaction, can be evaluated after implementing the security functions. However, the evaluation process plays a critical role in determining the usability level of the security requirement and discovering the usability issues as well.

When the usability experts complete the usability's evaluation process of the specified security requirements, they will decide whether these requirements have met the required usability level or not. Where if the identified usability requirements have not been achieved, the usability experts will clarify and explain to the security experts the usability issues that resulted from the specified security requirements. Hence, they will suggest recommendations that will assist in mitigating or resolving such issues, security experts in return should understand and analyze these recommendations and try to find a good compromise for improving the usability issues without breaching the security.

V. Case Study

A call center company has 2 branches, one is in Jordan and the other one is in Dubai. All the employees use their emails, which are related to the company, for sending their work reports to their managers, as well as, managers send messages to each other. The main manager of this company wants to develop a new website, one of the main functionalities that should be provided by the new website is encryption and decryption feature, such feature is aimed at allowing the company's employees to send and receive encrypted messages when they want to send sensitive information related to their company. This case study aims to clarify how the proposed trade-off model can be used in order to identify secure and usable requirements for such feature (see-Table 4).

V.1. Security requirement specification

Manuscript received November 2015, revised December 2015

In order to conduct this case study in an effective manner, we required help from a security expert and usability expert. As clarified in the proposed model (see Figure1), the security expert's responsibilities are constituted in identifying the security requirements specification of the encryption/decryption feature, in addition to conduct a risk assessment to assess the risk that may affect such feature, however, the security expert has specified the security requirements that are related to the encryption and decryption as the follows:

A. The users' messages should be encrypted using asymmetric encryption type.

B. When transferring the message from the client to the server, the communication channel should be encrypted using encryption protocol.

V.2. Risk assessment

As we previously mentioned, the risk assessment plays an important role in identifying the risks, their likelihood, and impact and then identifying the required actions to control it. We asked the security expert to identify in general the most critical risks, from his experienced background, which have high impact and likelihood, that may face the system from adding the messages encryption and decryption feature to the company's website. He clarified that the most critical risk that may face the messages when they are transmitted through the company's server is attacking the public key when the sender sends a request to the receiver to get his public key, the attacker may attack the message that contains the public key and change it to his public key, hence, he will be able to catch and decrypt the message that will be sent from sender to receiver. Thereby, the communication channel between the clients and company server should be encrypted by an encryption protocol (such as HTTPS protocol), he clarified other different risks and he suggested different actions that can be used to control these risks by identifying the right security policies.

V.3. Communication and coordination

In this step, we asked the security expert to describe the specified security requirements that are related to the user directly or indirectly in details. However, the introduced description was as follows:

"In order to encrypt and decrypt messages, each employee should have two keys, one called private and the other public, the private should be kept secret, each employee will select his private key, it should consist of at least 8 characters, and contains at least one number and symbol, it can't be retrieved if it's lost, the public key should be distributed to the other users. Each employee should create public and private keys and that is by filling a form, this form requires the employee to fill out their names, emails and entering and repeating a password, then the system will generate a file to be downloaded on the employee device, this file is called public key. If the employee wants to send an encrypted email to another employee, he should ask him to send his public key to receive the message. For example, if the sender wants to send encrypted message to the specific recipient, he will enter his private key and the recipient's public key, the message will be sent through encrypted channel, when the recipient receives the encrypted message, he will use his private key to decrypt it".

V.4. Specifying the usability requirements

We have already identified the usability requirements for the encryption/decryption requirement (see Appendix 1) , but for more efficiency, we gave the usability expert the description which is introduced by the security expert, and we asked him to identify the usability requirements of such description, he suggested that the following usability requirements should be achieved through the introduced description.

TABLE 4
USABILITY AND SECURITY EXPERTS SCENARIOS

Usability requirements		Description
Appropriateness recognizability	Description completeness	The system description should include the messages encryption/decryption feature.
	Understandable I/O	Users who don't have any previous experience in using the encryption/decryption function, in their first attempts: • When they want to create public and private keys, 90% of them must be able to clearly understand 1) the password's structure as an input within average 1 minute, 2) the generated public key file as an output within average 2 minutes; • When they want to send encrypted messages, 90% of them must be able to clearly understand 1) the public key as an input within average 1.5 minutes, 2) the resulted encrypted message as an output within average 1 minute; • When the users receive encrypted messages: 90% of them must be able to clearly understand 1) the encrypted messages that were received within average 1 minute, 2) the password as an input in order to transfer the messages to their original forms within average 40 seconds. Users who have previous experience in using the encryption/decryption function, in their first attempts: • When they want to create public and private keys, 90% of them must be able to clearly understand 1) the password's structure as an input within average 30 seconds, 2) the generated public key file as an output within average 1 minute; • When they want to send encrypted messages, 90% of them must be able to clearly understand 1) the public key as an input within average 40 seconds, 2) the resulted encrypted message as an output within average 30 seconds; • When the users received encrypted messages: 90% of them must be able to clearly understand 1) the received encrypted

	messages within average 30 seconds, 2) the password as an input in order to transfer the messages to their original forms within average 20 seconds. •

TABLE 4
USABILITY AND SECURITY EXPERTS SCENARIOS (Contd)

Usability requirements		Description
Learnability	Completeness of user documentation	The processes of keys generation, send encrypted messages and decrypt the encrypted messages steps should be described in details with images for clarification in the user documentation.
	Easy of help access	A link named help me should be added to help users completing the keys generation, send encrypted messages, or decrypt encrypted messages correctly, where the link should clarify with images how to complete each process, in addition to add an option to communicate with the system's administrator in order to resolve the problems that the users face through conducting the previous processes.
	Time to learn	• Users who don't have any previous experience in using the encryption/decryption function, in their first attempts, 1) 90% of them must be able to generate keys within 4 minutes, 2) 90% of them must be able to send encrypted message within 4 minutes, 3) 90% of them must be able to decrypt encrypted message within 2 minutes; • Users who have previous experience in using the encryption/decryption function, 1) 90% of them must be able to generate keys within an average of 2 minutes, 2) 90% of them must be able to send encrypted message within an average of 3 minutes, 3) 90% of them must be able to decrypt encrypted message within an average of 1 minutes.
	Memorability	Users who have not used the website for 5 days, they should be able to: 1) remember the steps that are required to send encrypted message within 2 minutes, 2) remember the steps that are required to decrypt encrypted message within an average 1/2 minute, 3) remember the password that is required to decrypt and encrypt messages within an average 1 minute.
User interface aesthetics	Appearance customizability of user interface	Users must be able to customize the theme of the encryption and decryption pages' background.
User error protection	Avoidance of incorrect operation	When generating the keys, the system must prevent users from lifting any field empty in the keys generation form, 2) the system must prevent users from using wrong passwords during the encryption and decryption process, 3) when sending encrypted message, the system must prevent users from lifting the recipient's public key empty.
	User error recoverability	In case that the users have encrypted messages with wrong public keys, they must be able to edit the messages before sending them, 2) in case that the users have forgotten their password, they must be able to recover them.
Operability	Operation consistency: Understandable system messages	Both the encryption and decryption processes should be accomplished in a consistent manner. • When the users fill the keys generation form: 1) if they forgot to fill any field, the system should provide a notification message to fill the empty field, 2) if the users have filled any field in a wrong manner, such as choosing password that consists of less than 8 elements, the system should provide an error message that clarifies the required password structure and ask the user to choose another one, 3) when the users send an encrypted message, the system should provide a notification message to clarify that the encrypted message is sent through a protected channel; • When the users want to send encrypted messages, 1) if they forget to enter the encryption password or they enter a wrong password, the system should provide an error message that clarifies that they should enter their passwords correctly, 2) if the users forget to add the recipient public key, the system should provide an error message that clarifies to users that they should enter the recipient's public key. •
Universality	Cultural universality	Users must be able to encrypt and decrypt text written in Arabic and English languages.

V.5. Evaluation

At this step, we asked the usability expert to evaluate each usability requirement specification through the description that is introduced by the security expert. The usability expert clarified that there are some usability requirements that can be easily achieved, such as description completeness, easy of help access, avoidance of incorrect operation, and some elements of understandable I/O. For example, regarding the understandable I/O, the structure of the required password can be easily clarified by adding a hint beside the password field, the resulted encrypted message or the received decrypted message can be easily clarified by adding a hint before the encrypted or decrypted message, such as: "this is an encrypted message, to read it, please click on the decrypt button".

Moreover, other usability requirements can't be easily achieved, where it will be difficult for the users, with their different knowledge and background, to understand and handle the public keys as an input and output, however, making users deal with such keys will increase the possibilities of producing errors. In addition, using at least 8 characters as a password with at least on number and one symbol may raise memorability's issues, moreover, too many actions are required to complete the encryption process, which will make this process hard to remember when they don't use the system for a long

period of time. Finally, users may forget their password and they should be able to recover it.

V.6. Suggest recommendations

We asked the usability expert to introduce recommendations that can assist in improving the usability, the following points clarified the introduced recommendations:

- Automating the public key management, where the user will not be required to handle them directly.
- Decreasing the password length and constraints, the user may not be able to easily remember a complicated password.
- Allowing users to recover their password if they lose them.

After introducing these recommendations to the security expert, he introduced different solutions for the introduced recommendations. Regarding the first recommendation, the security expert has suggested two different solutions with different security levels, where in the first solution, once the users generate their keys, the system will automatically store the user's public key with his email in a trusted key server. The security level of this solution depends on the security level of the used key server. The second solution aims to provide more security level by storing the users' public keys in the company server, where the security expert argued that the server can be protected in a higher manner using different ways, one of these ways is protecting the server using a firewall device, creating SQL database on the server in order to store the public keys on it, the connection type of this database should be local, where only the server can access it. In the two solutions, when the sender encrypts the message, the system will automatically use the recipient's email in order to request his public key from the key server or the company server, the public keys will be sent and received through an encrypted channel. However, the two solutions improve the usability level, where the user will not need to handle the keys directly.

Regarding the second recommendation, the security expert clarified that such step will affect the quality of the encryption process, and the password should be complex for a better security level. However, after clarifying that to the usability expert, he suggested that when the user generates his keys, it is possible to allow him to select an image that will appear each time he will enter his password, where such image may assist him to remember his password.

Finally, regarding the final recommendation, the security expert clarified that storing the passwords in the company server will not be secure, but in case the user loses his password, he can communicate with the system administrator, and prove his identity and the administrator will allow him to regenerate his keys, and replace the saved public key with the new one.

VI. Conclusion

Security and usability are widely recognized as two Quality requirements that should be handled at early development phases. In the current practices, the development teams usually focus on the security requirements in order to develop secure systems, in contrast of the usability requirements, which are mostly handled at later stages to be added as add-on features. Such behavior will lead to develop secure systems that can't be easily used by their users. The main objective of this paper is to introduce a trade-off model that can assist in balancing both security and usability requirements in order to develop systems with an adequate security and usability level. The objective was achieved by identifying the security and usability requirements, analyzing the relation between these requirements and finally designing the proposed security and usability requirements trade-off model.

The proposed trade-off model assists in identifying and specifying the usability requirements of each specified security requirement. However, as many researchers, we believe that, in many cases, the security and usability should be balanced, where improving one can affect the other. On the other hand, in some cases the usability requirements may serve the security requirements. Our proposed model does not limit on handling the usability requirements that are required to be balanced with the security requirements, it also involves identifying and evaluating the usability requirements that are not required to be balanced with the security requirements, which in return assist in improving the usability level of the identified security requirements.

References

[1] K. T. Al-Sarayreh, "Identification, specification and measurement, using international standards, of the system non functional requirements allocated to realtime embedded software," École de technologie supérieure, 2011.

[2] Abran, A., K. T. Al-Sarayreh, and J. J. Cuadrado-Gallego, "A Standards-based Reference Framework for System Portability Requirements", *Computer Standards and Interface, Elsevier*, V:35(4):pp.:380-395, 2013. http://dx.doi.org/10.1016/j.csi.2012.11.003

[3] Al-Sarayreh, K. T., A. Abran and J. J. Cuadrado-Gallego, " A Standards-based model of system maintainability requirements", *Journal of Software: Evolution and Process*, John Wiley & Sons, Ltd, V:25(5): pp:459-505, 2013. http://dx.doi.org/10.1002/smr.1553.

[4] Khalid T. Al-Sarayreh, " Dependability Model for Decomposition and Allocation of System Safety Integrity Levels of Software Quality", *International Review on Computers and Software (I.RE.CO.S)*, Vol. 10, No.12, 2015.

[5] H. S. Jabbar, T. V. Gopa," An Integrated Metrics Based Approach for Usability Engineering", *International Journal on Information Technology (IREIT)*, Vol. 2, No 5, 2015.

[6] Julia, S., Rodrigues, P., "Analyzing Software Architecture Styles Using Quality Attributes", *International Journal on Information Technology (IREIT)*, Vol. 2, No. 2, 2014.

[7] Bureš, V. "Comparative Analysis of System Dynamics Software Packages", *International Review on Modelling and Simulations (IREMOS)*, Vol. 8, No.2, 2015.

[8] Khalid T. Al-Sarayreh, Asma Labadi and Kenza Meridji, "A Generic Method for Identifying Maintainability Requirements Using ISO Standards", *International Conference on Intelligent Information Processing, Security and Advanced Communication (IPAC 2015)*, ISBN: 978-1-4503-3458-7, ACM, ICPS, Algeria, NOV. 23-25, 2015.

[9] Lina Hasan and Khalid T. Al-Sarayreh, "An Integrated Measurement Model for Evaluating Usability Attributes", *International Conference on Intelligent Information Processing, Security and Advanced Communication (IPAC 2015)*, ISBN: 978-1-4503-3458-7, ACM, ICPS, Algeria, NOV. 23-25, 2015.

[10] Khalid T. Al-Sarayreh and Kenza Meridji, "Towards a Development of an Operational Process for Software Requirements: Case study application for Renewable Energy Software", *International journal of software engineering and its applications*. IJSEIA. Vol. 9, No. 7, pp. 11-26, 2015.

[11] Kenza Meridji, Khaled AlMakadmeh , Khalid T. Al-Sarayreh, Anas Abuljadayel and Mohammad Khalaf, "Towards A Requirements Model of System Security Using International Standards", *International journal of software engineering and its applications*. IJSEIA. Vol. 9, No. 4, pp. 139-164, 2015.

[12] J. Zhou and D. Gollmann, "Observations on non-repudiation," in *Advances in Cryptology—ASIACRYPT'96*, 1996, pp. 133-144.

[13] R. Gajanayake, R. Iannella, and T. Sahama, "Sharing with care: An information accountability perspective," *Internet Computing, IEEE*, vol. 15, pp. 31-38, 2011.

[14] Y. Sattarova Feruza and T.-h. Kim, "IT security review: Privacy, protection, access control, assurance and system security," *International journal of multimedia and ubiquitous engineering*, vol. 2, pp. 17-31, 2007.

[15] D. Zissis and D. Lekkas, "Addressing cloud computing security issues," *Future Generation computer systems*, vol. 28, pp. 583-592, 2012.

[16] M. Agrawal and P. Mishra, "A comparative survey on symmetric key encryption techniques," *International Journal on Computer Science and Engineering (IJCSE)*, vol. 4, pp. 877-882, 2012.

[17] K. T. Fung, *Network security technologies*: CRC Press, 2004.

[18] M. Ciampa, *Security+ guide to network security fundamentals*: Cengage Learning, 2011.

[19] R. B. Grady, *Practical software metrics for project management and process improvement*: Prentice-Hall, Inc., 1992.

[20] J. Nielsen, *Usability engineering*: Elsevier, 1994.

[21] A. Abran, A. Khelifi, W. Suryn, and A. Seffah, "Usability meanings and interpretations in ISO standards," *Software Quality Journal*, vol. 11, pp. 325-338, 2003.

[22] ISO, "IEC25022: 2012 - Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) – Measurement of quality in use," *International Organization for Standardization*, 2012.

[23] A. Seffah, M. Donyaee, R. B. Kline, and H. K. Padda, "Usability measurement and metrics: A consolidated model," *Software Quality Journal*, vol. 14, pp. 159-178, 2006.

[24] D. Gupta, A. Ahlawat, and K. Sagar, "A critical analysis of a hierarchy based Usability Model," in *Contemporary Computing and Informatics (IC3I), 2014 International Conference on*, 2014, pp. 255-260.

[25] K. T. Al-Sarayreh, I. Al-Oqily, and K. Meridji, "A standard-based reference framework for system operations requirements," *Int. J. Comput. Appl. Technol.*, vol. 47, pp. 351-363, 2013.

[26] M. Rausand, *Risk Assessment: Theory, Methods, and Applications*: Wiley, 2013.

Dr. Khalid T. Al-Sarayreh is an assistant professor of Software Engineering at Hashemite University in Jordan since 2011. He has a PhD degree in Software Engineering from École de Technologie Supérieure (ÉTS) – University of Québec in Montreal, Canada. He also has a Doctoral degree in Computer Information Systems, MSc in Computer Engineering (Embedded Systems) and BS degree in Computer Science from Jordanian Universities. From 2002 to 2005, he was at the KADDB (King Abdullah II Design and Development Bureau). From 2005 to 2006, he was a lecturer at the Jordan University, and from 2006-2008, he was an Assistant Professor at the Faculty of Information Technology at the Applied Sciences University (Jordan).
His research interests include: Software Quality Engineering, Software Measurement, Function Points, Non functional requirements for Embedded Systems and software, Software Reuse, Software Engineering Standards (ECSS, IEEE and ISO) [khalidt@hu.edu.jo].

Lina Hasan is a Master student in the Software Engineering Department at The Hashemite University, Jordan. She completed his B.S. in Software Engineering from The Hashemite University, Jordan in 2012. Her research interest is Software Engineering with focus on Software Requirements and Software Quality.

Khaled Almakadmeh is an Assistant Professor in the Software Engineering Department at Hashemite University, Jordan. He completed his Ph.D. in Software Engineering at University of Quebec (2013). He received his Master in Information Systems Security from Concordia University (2010) and his B.S. in computer science from Jordan University of Science & Technology (2008). His research interest include software requirements, software metrics and software quality.

Appendix 1

Usability requirements	Security requirements												
	Authenticity		Confidentiality			Non-Repudiation		Accountability	Data Integrity		Availability		
	Authentication protocols and rules		Encryption/ decryption		Authori-zation	Digital signature		Audit logs	Hash function	Backup	Network redundancy	Auto-matic restart	Auto-matic backup
	Requires UI	Doesn't require UI	Requires UI	Doesn't require UI		SDD	SPPK						
Learnability	√	√	√	√	√	√	√	√		√			
Appropriateness recognizability	√	√	√	√	√	√	√	√	√	√	√	√	√
User error protection	√		√		√	√	√	√		√			
Operability	√	√	√	√	√	√	√	√	√	√	√	√	√
User interface aesthetics	√		√			√	√	√		√			
Universality	√		√				√	√					
Effectiveness	√		√			√	√	√		√			
Efficiency	√		√			√	√	√					
Productivity			√			√	√	√		√			
Satisfaction	√		√			√	√	√		√			
- UI: User Interaction – SDD: Sign Digital Document –SPP: Sign using Public and Private Keys													