ICIC Express Letters
Part B: Applications
Volume **10**, Number **8**, August **2019**

ICIC International ©2019 ISSN 2185-2766
pp. **665–672**

USABLE-SECURITY ASSESSMENT THROUGH A DECISION MAKING PROCEDURE

Alka Agrawal[1], Mamdouh Alenezi[2], Dhirendra Pandey[1], Rajeev Kumar[1,*] and Raees Ahmad Khan[1]

[1]Department of Information Technology
Babasaheb Bhimrao Ambedkar Central University
Lucknow, Uttar Pradesh 226025, India
alka_csjmu@yahoo.co.in; prof.dhiren@gmail.com; *Corresponding author: rs0414@gmail.com
khanraees@yahoo.com

[2]College of Computer and Information Sciences
Prince Sultan University
P.O.Box No. 66833 Rafha Street, Riyadh 11586, Saudi Arabia
malenezi@psu.edu.sa

Received December 2018; accepted March 2019

Abstract. *Security has always been a vital research topic since the birth of software. A great deal of research has been conducted to determine the ways for identifying and classifying the security issues or goals. Unfortunately, the highly secure design of software becomes worthless because the usability of services is very low. Furthermore, usable-security is in much demand due to high investment in recent years. To improve the usability of security services, there is a need to focus on usability along with security. Usable-security attributes have their own impact while integrating security, usability, and assessment of usable-security plays a crucial role during software development. Within this context, this paper estimates the usable-security of the two alternatives versions of the software called version 1 and version 2. To assess the usable-security, authors are using the Fuzzy Analytic Hierarchy Process (Fuzzy AHP) methodology. In addition, the impact of the security on usability and impact of the usability on security are evaluated quantitatively.*

Keywords: Software security, Software usability, Usability of security services, Software development process

1. **Introduction.** Major quality factors including maintainability, usability, and security etc., are always considered during software development [1]. Nowadays, developers are facing usability related problem after delivering the software to end users [2]. Due to high-security design, software is not usable as it could be [1-3]. Practitioners are trying to find a solution to this problem. Usability of software increases, if security is usable [4,5]. To achieve the high user's satisfaction and business continuity, a perfect balance between usability and security is essential after delivering the software to end users [2] because increasing security of software decreases the usability [3]. Prevention of un-authorization is the main aim of security while usability focuses on the ease of users 'keeping simple' formula [2,3]. The organization should focus on ensuring usability while preserving the rights and security of software services.

In this row, practitioners are trying to assess and improve the usability of security services, quantitatively [6]. Unfortunately, there is very little literature available related to usability of security or usable-security. In addition, the user is the person who authorizes the security settings; hence usable and secure services are the need of today's generation [7,8]. Also attributes of security and usability play an important role in the assessment

DOI: 10.24507/icicelb.10.08.665

of the usable-security of software services [9]. Usable-security of software services may be affected by usability and security attributes including Effectiveness, Efficiency, and Satisfaction (EES) [10] & Confidentiality, Integrity, and Availability (CIA) [11]. These attributes contribute to assessing and improving the usable-security of software services. The assessment of usable-security should not focus only on security but it is important for the whole usability of software services.

Rest of the paper is organized as follows: the second section discusses the introduction of usable-security of software services, the third section is describing the assessment method, i.e., Fuzzy AHP, implements the method and obtains the results. The findings and the final conclusion are presented in Section four.

2. **Usability of Security Services.** Software security is an idea or method that is implemented to prevent software from malicious attacks [12-14]. According to G. McGraw, software security is about building secure software, i.e., designing software to be secure, making sure that the software is secure, and educating software developers and architects, and users about how to build secure software [15,16]. Due to the wide application of software, security has become a key part of the software development process [13]. In fact, software faces threats from various potential malicious adversaries that grow every day, from web-conscious PC applications to complex media communications [14,17].

Security and usability practitioners must learn to work together on both the topics to create a very well secure software [13,14]. Although it seems to be found odds as usability and security have an inversely proportional relationship between them. It is revealed that improving one of them affects the other. Techniques to incorporate security issues or goals have already been developed [15], but there is missing an important aspect, i.e., security-usability/usable-security. Usability in the security must be incorporated into usable security from the very beginning and it should be continued till the security services are running [16,17]. The International Standard Organisation (ISO) [7] defines usability as the ability that provides specified services with ease of use to the user including effectiveness, efficiency, and satisfaction in a specified context of use. According to this definition, usable-security focuses on the user's goals (effectiveness), the speed with which goals are achieved (efficiency), and users' satisfaction.

Hence, security has three major factors of usability that affect indirectly including effectiveness, efficiency, and satisfaction. Further, CIA is the pillars of security [17]. This work contributes as an assessment of usable-security through Fuzzy AHP. In this work, authors have taken six factors of usable-security including confidentiality, integrity, availability, effectiveness, efficiency, and satisfaction as shown in Figure 1.

Fuzzy AHP is chosen for assessing the usable-security because it is capable of controlling vague judgmental inputs given by the participants [18-20]. It is also capable of converting qualitative inputs into quantitative results, in the form of weight and ranking which is

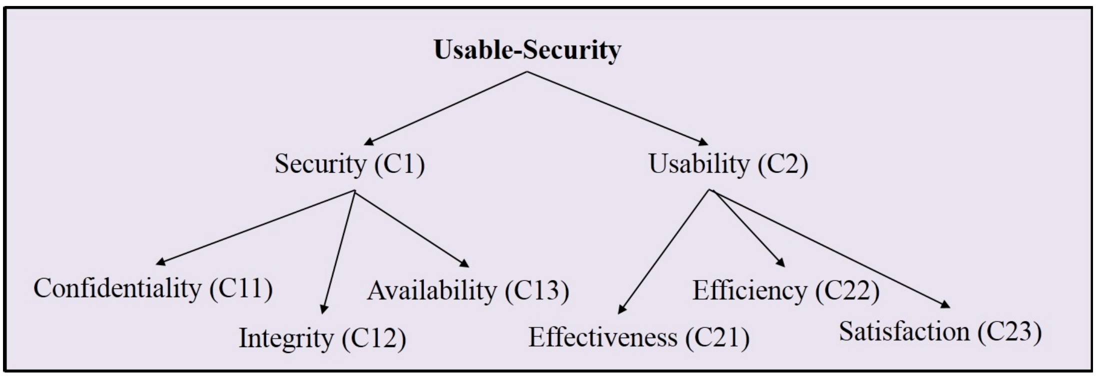

Figure 1. Tree structure of usable-security

an effective assessment of usable-security [21-23]. To assess the usable-security through analyzing data and reaching a consensus among experts, this work adopts the Buckley method [18] and method of the eigenvector is used to evaluate the weights. The Fuzzy AHP method comprises four major steps as discussed below.

The first step is establishing the hierarchical structure on the basis of the problem. The problem should be stated clearly and decomposed into a rational system like a hierarchy [24]. The structure can be determined by the expert's opinions through brainstorming or other appropriate methods. The second step is establishing the triangular fuzzy numbers. Fuzzy set theory is able to handle vague data. A TFN is denoted simply as (L, M, U). Equations (1)-(3) are used to convert the numeric values into Triangular Fuzzy Number (TFN) [19] and denoted as (L_{ij}, M_{ij}, U_{ij}) where L_{ij} is least possible, M_{ij} is most likely and U_{ij} is extreme possible events. Further, TFN $[\eta_{ij}]$ is established as the following:

$$\eta_{ij} = [L_{ij}, M_{ij}, U_{ij}] \quad \text{where } L_{ij} \leq M_{ij} \leq U_{ij}$$

$$L_{ij} = \min(J_{ijk}) \tag{1}$$

$$M_{ij} = (J_{ij1}, J_{ij2}, \ldots, J_{ijk})^{1/k} \tag{2}$$

$$U_{ij} = \max(J_{ijk}) \tag{3}$$

In the above equations, J_{ijk} shows the comparative importance of the values between two criteria and given by expert k, where i and j represent a pair of criteria being judged by experts. Further, after the construction of the comparison matrix, defuzzification is performed to produce a quantifiable value based on the calculated TFN values with the help of Equations (4)-(6) that is called alpha cut method [19-22].

$$\mu_{\alpha,\beta}(\eta_{ij}) = [\beta \cdot \eta_\alpha(L_{ij}) + (1-\beta) \cdot \eta_\alpha(H_{ij})] \tag{4}$$

where $0 \leq \alpha \leq 1$ and $0 \leq \beta \leq 1$, such that,

$$\eta_\alpha(L_{ij}) = (M_{ij} - L_{ij}) \cdot \alpha + L_{ij} \tag{5}$$

$$\eta_\alpha(H_{ij}) = H_{ij} - (H_{ij} - M_{ij}) \cdot \alpha \tag{6}$$

where α and β in these equations are used for the preferences of experts. These two values vary between 0 and 1. Crisp sets $\rho_{\alpha,\beta}(\tilde{A})$ simply describe whether an element is either a member of the set or not. Single pair-wise comparison matrix is expressed in Equation (7).

$$\rho_{\alpha,\beta}\left(\tilde{A}\right) = \rho_{\alpha,\beta}[\tilde{a}_{ij}] = \begin{array}{c} \\ C_1 \\ C_2 \\ \vdots \\ C_n \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & \cdots & C_n \end{array} \\ \begin{bmatrix} 1 & \rho_{\alpha,\beta}(\tilde{a}_{11}) & \cdots & \rho_{\alpha,\beta}(\tilde{a}_{1i}) \\ 1/\rho_{\alpha,\beta}(\tilde{a}_{21}) & 1 & \cdots & \rho_{\alpha,\beta}(\tilde{a}_{2i}) \\ \vdots & \vdots & & \vdots \\ 1/\rho_{\alpha,\beta}(\tilde{a}_{j1}) & 1/\rho_{\alpha,\beta}(\tilde{a}_{j2}) & \cdots & 1 \end{bmatrix} \end{array} \tag{7}$$

The next step is to determine the eigenvalue and eigenvector of the pairwise comparison matrix. Assume that μ denotes the eigenvector while λ denotes the eigenvalue of fuzzy pairwise comparison matrix η_{ij}.

$$[\mu_{\alpha,\beta}(\eta_{ij}) - \lambda I] \cdot \mu = 0 \tag{8}$$

Equation (8) is based on the linear transformation of vectors, where I represents the unitary matrix. By applying Equations (1)-(8), the weights of particular criteria with respect to all other possible criteria may be acquired.

3. **Assessment of Usable-Security.** AHP is considered good in analyzing a decision in a group, but many researchers have found that Fuzzy AHP is more valuable to provide crisp decisions with their weights too [19-22]. In addition, it has been an important tool that is widely used to complete priority analysis and adopted by decision makers. To deal with the uncertainties and ambiguity of human judgment, the authors took a modified version of AHP known as Fuzzy AHP [23]. For collecting data, authors have taken 50 experts from the different fields of academics and industry. With the help of the inputs of experts, this contribution aims to evaluate the usable-security. To evaluate the usable-security, two versions of a software including version 1 and version 2 have been taken. To assess the best alternative, Figure 1 shows the hierarchy of the usable-security attributes.

Further, Equations (1)-(3) are used to evaluate the triangular fuzzy numbers. The constructed aggregated fuzzy pair-wise comparison matrix prepared by the researchers after evaluating judgments from twenty participants is shown in Table 1 to Table 3.

Table 1. Aggregated pair-wise comparison matrix at level 1

	Security (C1)	**Usability (C2)**
Security (C1)	1, 1, 1	0.69, 0.89, 1.10
Usability (C2)	–	1, 1, 1

Table 2. Aggregated pair-wise comparison matrix for security at level 2

	Confidentiality (C11)	**Integrity (C12)**	**Availability (C13)**
Confidentiality (C11)	1, 1, 1	0.66, 1.17, 1.69	0.70, 0.95, 1.35
Integrity (C12)	–	1, 1, 1	1.19, 1.58, 2.15
Availability (C13)	–	–	1, 1, 1

Table 3. Aggregated pair-wise comparison matrix for usability at level 2

	Effectiveness (C21)	**Efficiency (C22)**	**Satisfaction (C23)**
Effectiveness (C21)	1, 1, 1	0.23, 0.28, 0.36	1.15, 1.44, 1.70
Efficiency (C22)	–	1, 1, 1	0.31, 0.39, 0.56
Satisfaction (C23)	–	–	1, 1, 1

After constructing fuzzy pair-wise comparison matrices, Equations (4)-(6) are used for defuzzification. Further, Equation (7) is used to evaluate the single pair-wise comparison matrix (Crisp set). The next step is to determine the eigenvalue and eigenvector of the fuzzy pairwise comparison matrix. By applying Equations (1)-(8), the weights of particular criteria with respect to all other possible criteria may be acquired that are shown in Table 4 to Table 6 and Figure 2 to Figure 4.

Table 4. Defuzzified matrix and local weights for usable-security at level 1

	Security (C1)	Usability (C2)	Weights
Security (C1)	1	0.89	**0.471**
Usability (C2)	1.1236	1	**0.529**
			C.R. = 0.002

Table 5. Defuzzified matrix and local weights for security at level 2

	Confidentiality (C11)	Integrity (C12)	Availability (C13)	Weights
Confidentiality (C11)	1	1.17	0.99	**0.348**
Integrity (C12)	0.85	1	1.63	**0.370**
Availability (C13)	1.01	0.61	1	**0.282**
				C.R. = 0.023

Table 6. Defuzzified matrix and local weights for usability at level 2

	Effectiveness (C21)	Efficiency (C22)	Satisfaction (C23)	Weights
Effectiveness (C21)	1	0.29	1.36	**0.238**
Efficiency (C22)	3.45	1	0.41	**0.366**
Satisfaction (C23)	0.74	2.44	1	**0.396**
				C.R. = 0.035

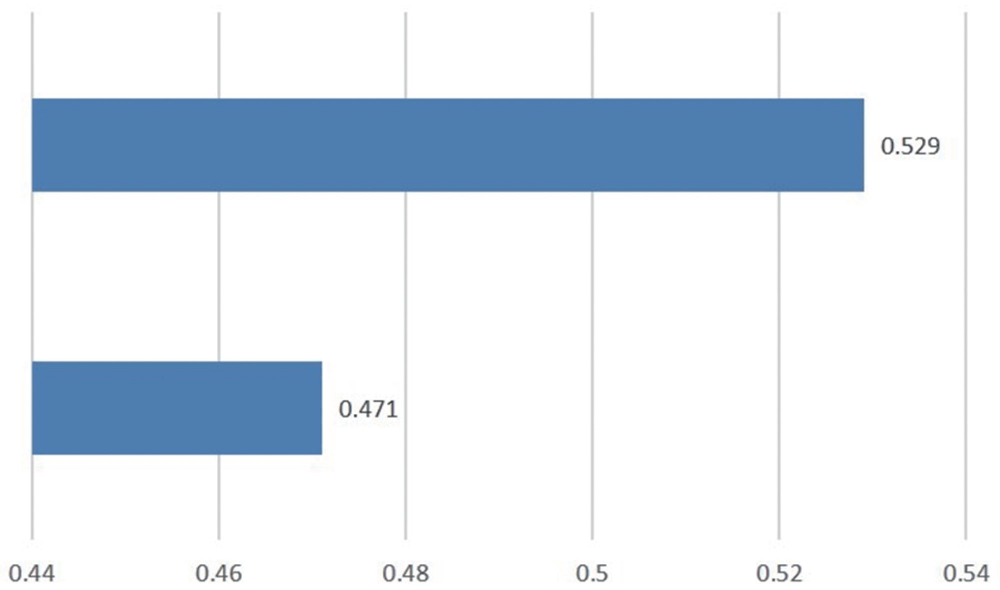

Figure 2. Local weights for usable-security at level 1

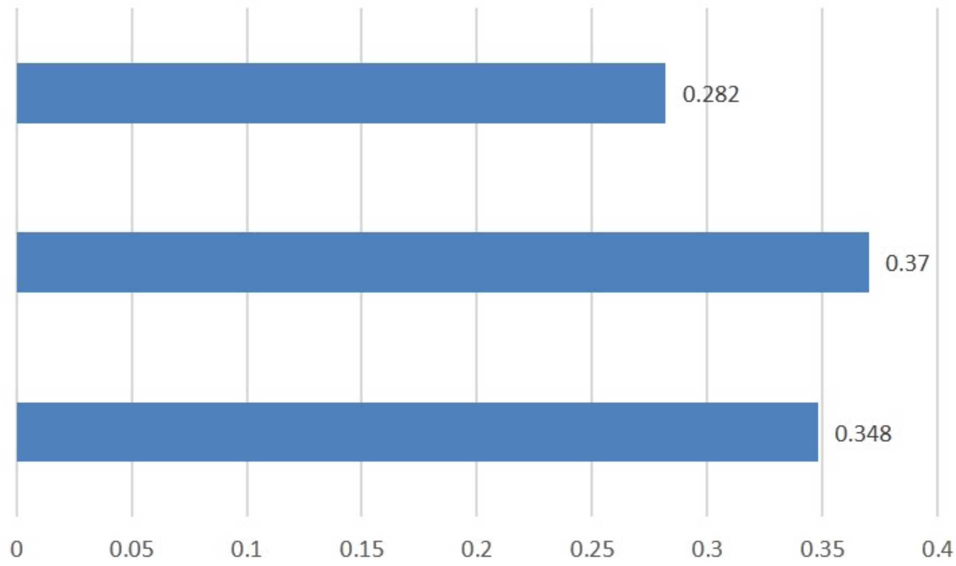

FIGURE 3. Local weights for security at level 2

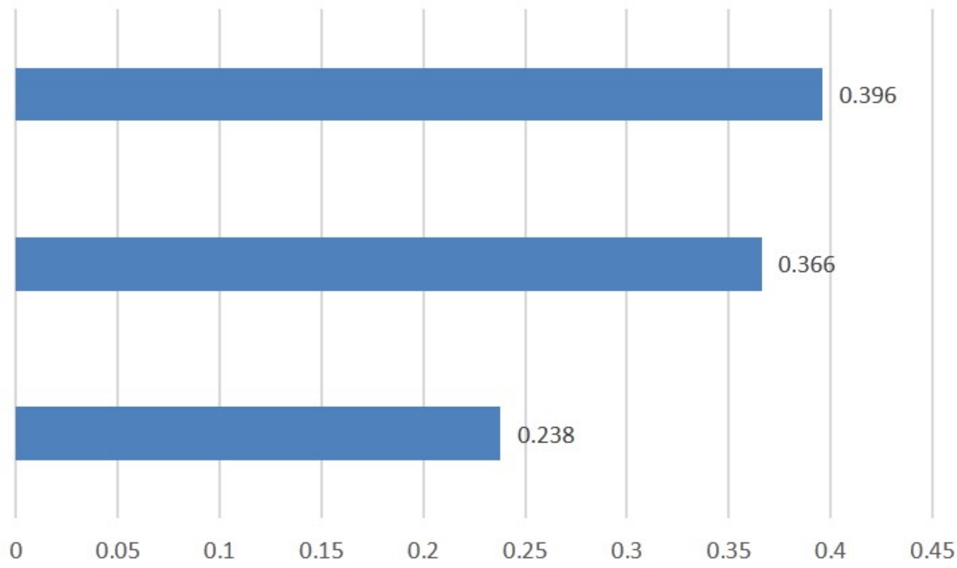

FIGURE 4. Local weights for usability at level 2

The composite priorities of levels 2 to 3 are then determined by aggregating the weights throughout the hierarchy. The six evaluative criteria are weighed as follows: confidentiality (0.164), integrity (0.174), availability (0.133), effectiveness (0.126), efficiency (0.194), satisfaction (0.209) and satisfaction is most important for usable-security of software services. Table 7 summarizes eigenvector results for levels 1 and 2 and final weights of usable-security attributes are shown in Figure 5 and usable-security is determined as [21]:

$$[0.164, 0.174, 0.133, 0.126, 0.194, 0.209] \begin{bmatrix} 0.25 & 0.27 \\ 0.24 & 0.25 \\ 0.27 & 0.25 \\ 0.33 & 0.24 \\ 0.25 & 0.21 \\ 0.23 & 0.37 \end{bmatrix} = \begin{matrix} \text{Version 1} \\ \text{Version 2} \end{matrix} = \begin{bmatrix} \mathbf{0.257} \\ \mathbf{0.270} \end{bmatrix}$$

Table 7. Overall results

The first level attributes	Local weights of the first level	The second level attributes	Local weights of the second level	Overall weights	Overall ranks	Weights for level 3	
						Version 1	Version 2
C1	0.471	C11	0.348	0.164	4	0.25	0.27
		C12	0.370	0.174	3	0.24	0.25
		C13	0.282	0.133	5	0.27	0.25
C2	0.529	C21	0.238	0.126	6	0.33	0.24
		C22	0.366	0.194	2	0.25	0.21
		C23	0.396	0.209	1	0.23	0.37

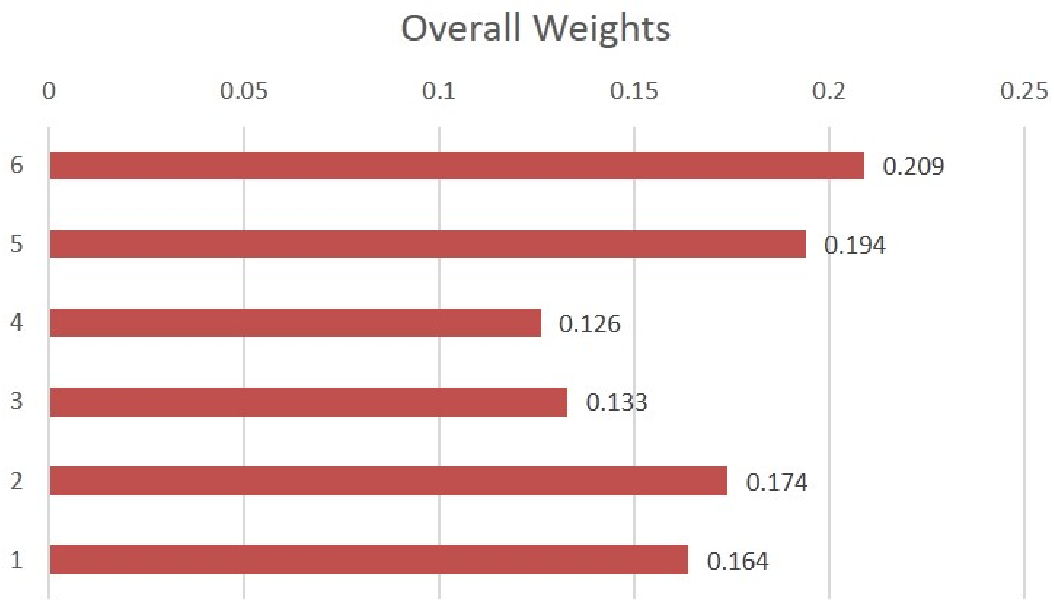

Figure 5. Absolute weights of attributes for usable-security

Usable-security of version 2 is higher than version 1, i.e., development of version 2 of software is appropriate. Based on these results, the conclusion focuses on providing suggestions to developers for enhancing the effectiveness and efficiency of security usability of software services. In actual scenario, there are various usable-security attributes, which are present in the software development process [6]. In this research, only six attributes of usable-security, which affect security have been identified as well as assessed.

4. **Conclusion.** The different security models are helpful to generate quantitative values including object-oriented and service-oriented perspective but there is no such measure available, which can measure security-usability. The model proposed here will help to evaluate the usable-security of software services and satisfaction of the user. In this research, usability and security attributes are identified and usable-security of software services is examined. For the assurance of usable-security, developers need to firstly focus on availability for ensuring usable-security and software services.

Acknowledgment. Authors are thankful to College of Computer and Information Sciences, Prince Sultan University for providing the fund to carry out the work.

REFERENCES

[1] S. Ruoti, B. Roberts and K. Seamons, Authentication melee: A usability analysis of seven web authentication systems, *Report of the International World Wide Web Conferences Steering Committee*, pp.916-926, 2015.

[2] Y. Liu, Analyzing Facebook privacy settings: User expectations vs. reality, *ACM SIGCOMM*, 2011.

[3] J. Neilson, Factors and principles affecting the usability of four e-commerce sites, *Proc. of the 4th Conference on Human Factors & the Web*, Basking Ridge, NJ, 1988.

[4] R. Kumar, S. A. Khan and R. A. Khan, Durability challenges in software engineering, *The Journal of Defense Software Engineering*, vol.42, no.4, pp.29-31, 2016.

[5] R. S. Pressman, *Software Engineering: A Practitioner's Approach*, Palgrave Macmillan, London, 2005.

[6] I. Fléchais, *Designing Secure and Usable Systems*, Ph.D. Thesis, University College London, 2005.

[7] ISO 9241-11:1998, *Ergonomic Requirements for Office Work with Visual Display Terminals*, The International Organization for Standardization, Geneva, 1998.

[8] G. McGraw, Software assurance for security, *IEEE Computer*, vol.32, no.4, pp.103-105, 1999.

[9] *The Importance of Usability for Secure Software*, https://www.signiant.com/blog/the-importance-of-usability-for-secure-software/, 2018.

[10] *Usability in Software Design*, https://docs.microsoft.com/en-us/windows/desktop/appuistart/usability-in-software-design, 2018.

[11] A. Whitten, *Making Security Usable*, Ph.D. Thesis, Carnegie Mellon University, 2004.

[12] N. S. Good and A. Krekelberg, Usability and privacy: A study of Kazaa P2P file sharing, *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*, pp.137-144, 2003.

[13] M. Anwar, M. S. A. Malik and A. Siddiq, Usability and security issues of the user interface design, *International Journal of Computer Science and Network Security*, vol.18, no.8, pp.105-111, 2018.

[14] O. Kulyk and M. Volkamer, *Usability Is not Enough: Lessons Learned from 'Human Factors in Security' Research for Verifiability*, E-Vote-ID 2018, 2018.

[15] J. Hansen, K. Porter, A. Shalaginov and K. Franke, Comparing open source search engine functionality, efficiency and effectiveness with respect to digital forensic search, *NISK 2018*, 2018.

[16] *Computer Hope*, http://www.computerhope.com/jargon/p/privacy.htm, 2018.

[17] J. H. Saltzer and M. D. Schroeder, The protection of information in computer systems, *Proc. of the IEEE*, vol.63, no.9, pp.1278-1308, 1975.

[18] B. Beckles, V. Welch and J. Basney, Mechanisms for increasing the usability of grid security, *International Journal of Human Computer Studies*, vol.63, no.12, pp.74-101, 2005.

[19] J. J. Buckley, Fuzzy hierarchical analysis, *Fuzzy Sets and Systems*, vol.17, no.3, pp.233-247, 1985.

[20] M. Alenezi, R. Kumar, A. Agrawal and R. A. Khan, Usable-security attribute evaluation using fuzzy analytic hierarchy process, *ICIC Express Letters*, vol.13, no.6, pp.453-460, 2019.

[21] H. Deng, Multi criteria analysis with fuzzy pair wise comparisons, *International Journal of Approximate Reasoning*, vol.21, pp.215-231, 1999.

[22] T. L. Saaty, *The Analytic Hierarchy Process*, McGraw Hill, New York, 1980.

[23] C. Chang, C. Wu and H. Lin, Integrating fuzzy theory and hierarchy concepts to evaluate software quality, *Springer Software Quality Journal*, vol.16, pp.263-276, 2008.

[24] T. S. Liou and M. J. J. Wang, Ranking fuzzy numbers with integral value, *Fuzzy Sets and Systems*, vol.50, no.3, pp.247-255, 1992.