applied sciences

MDPI

Article

The Evaluation of Software Security through Quantum Computing Techniques: A Durability Perspective

Hashem Alyami [1], Mohd Nadeem [2], Abdullah Alharbi [3], Wael Alosaimi [3], Md Tarique Jamal Ansari [2], Dhirendra Pandey [2], Rajeev Kumar [4,*] and Raees Ahmad Khan [2]

1 Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia; hyami@tu.edu.sa
2 Department of Information Technology, Babasaheb Bhimrao Ambedkar University, Lucknow 226025, Uttar Pradesh, India; mohd.nadeem1155@gmail.com (M.N.); tjtjansari@gmail.com (M.T.J.A.); profdhriendra@gmail.com (D.P.); khanraees@yahoo.com (R.A.K.)
3 Department of Information Technology, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia; amharbi@tu.edu.sa (A.A.); w.osaimi@tu.edu.sa (W.A.)
4 Department of Computer Science and Engineering, Babu Banarasi Das University, Lucknow 226028, Uttar Pradesh, India
* Correspondence: rs0414@gmail.com

Abstract: The primary goal of this research study, in the field of information technology (IT), is to improve the security and durability of software. A quantum computing-based security algorithm springs quite a lot of symmetrical approaches and procedures to ensure optimum software retreat. The accurate assessment of software's durability and security is a dynamic aspect in assessing, administrating, and controlling security for strengthening the features of security. This paper essentially emphasises the demarcation and depiction of quantum computing from a software security perspective. At present, different symmetrical-based cryptography approaches or algorithms are being used to protect different government and non-government sectors, such as banks, healthcare sectors, defense, transport, automobiles, navigators, weather forecasting, etc., to ensure software durability and security. However, many crypto schemes are likely to collapse when a large qubit-based quantum computer is developed. In such a scenario, it is necessary to pay attention to the security alternatives based on quantum computing. Presently, the different factors of software durability are usability, dependability, trustworthiness, and human trust. In this study, we have also classified the durability level in the second stage. The intention of the evaluation of the impact on security over quantum duration is to estimate and assess the security durability of software. In this research investigation, we have followed the symmetrical hybrid technique of fuzzy analytic hierarchy process (FAHP) and fuzzy technique for order of preference by similarity to ideal solution (FTOPSIS). The obtained results, and the method used in this estimation, would make a significant contribution to future research for organising software security and durability (SSD) in the presence of a quantum computer.

Keywords: software security; software durability; quantum computing; symmetrical technique; fuzzy AHP; fuzzy TOPSIS

check for updates

Citation: Alyami, H.; Nadeem, M.; Alharbi, A.; Alosaimi, W.; Ansari, M.T.J.; Pandey, D.; Kumar, R.; Khan, R.A. The Evaluation of Software Security through Quantum Computing Techniques: A Durability Perspective. *Appl. Sci.* **2021**, *11*, 11784. https://doi.org/10.3390/app112411784

Academic Editors: Seongsoo Cho and Bhanu Shrestha

Received: 20 November 2021
Accepted: 8 December 2021
Published: 11 December 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Software security and durability (SSD) in the development cycle of software has, at present, created new difficulties for developers [1]. Moreover, the astonishing extension of computing in the quantum period and tremendous improvement in programming have generated the necessity for building effective security mechanisms in the initial stages of software development itself. Thus, SSDs have become a vital element for software advancement [2]. Although the developers invest many resources in resolving security concerns throughout the initial phase of software development [3], no consideration is given to the

life span of the software. Software with limited or flawed security durability will come up short in an exceptionally serious market. Hence, software development associations ought to put huge resources into figuring out the precept of durability-security. From the software security point of view, software advancement incorporates security credits, security procedures, security plans, security testing, and security across the board. Software security is not foolproof [2]. The reason for this is that, despite the growing demand for secure software, developers are encountering new challenges in meeting users' demands while developing the product [4]. In addition, the software developers themselves face several challenges that include limitations in improvement because of cost, time-to-market necessities, profitability sway, consumer loyalty concerns, etc. The outcome of such issues has inappropriately evolved secure programming with low security [5]. Further, quantum security manages the retreat in the time of quantum registering. These days, the pace of advancement in quantum innovation is dramatic. A gathering of researchers has effectively fostered a completely quantum processor, the Sycamore processor, which can plan the quantum circuit in two hundred seconds. The equivalent would be created by an old-style supercomputer in ten thousand years [5]. With the advancement of quantum-based processors, the encryption- or balanced-based security techniques for various organizations, web applications, and software security, as well as all that depends on the computer network, is in question. The SSD is likewise impacted by quantum computers. The security of software is based on utilising the approach of the security key, which is a number according to the Shor calculation. The enormous number 2042 can be factored to its indivisible amount [6]. The entire time spent by a traditional computer can be factored without interruption by the long term.

In addition, quantum processing is a notable innovation in the field of IT that can uphold worldwide endeavours in tending to SSDs. As a result, software development firms are focusing on ensuring that the durability of their products development life cycle comprises of various stages, for instance, requirements for planning, planning, coding, testing, and investigation in conclusion support. Upkeep is considered the last period of headway [7]. The handiness of programming should be solid to achieve practicality. Shareware strength is a matter of time, during which the shareware works properly [8]. The advancement of processing in the twenty-first century makes programming and organisation shaky. The SSD is drawing the researchers' interests at present. Moreover, the developers are also working on mechanisms to strengthen the durability of software, along with security. The developers have characterised SSD as the duration during which the products are executed securely. To create more efficacious security strength, there is a need to examine the association between durability, its attributes, and security for secure SSD [9].

Further, appraisals of secure and durable software will also help organisations know the longevity of their products. Building an SSD is a complex apprehension; thus, SSD credits should be calculated cautiously, as they are significant devices of extensive security while utilising software [10]. To evaluate and advance the SSD strength, it is necessary to close the gap between quantum security and traditional computer security procedures in order to strengthen these characteristics. Security durability might be improved by estimating the significance of different affecting factors and alternatives. Furthermore, this paper also assesses the significance of the attributes of security and durability. This symmetrical methodology contributes to expanding the security strength of the software. The issue of evaluating security durability credits is a dynamic issue. Thus, the multi-criteria decision making (MCDM) strategy is the most appropriate means for evaluating the security durability. MCDM techniques can be utilised in several contexts, including software, frameworks, and many more [11]. The MCDM procedure permits the developers to choose options among various conflicting choices, especially when the specialists have doubts about their decisions [12]. The conflicts in the choices of specialists motivated us to utilise FAHP and FTOPSIS techniques, because fuzzy frameworks help in assessing ambiguous and uncertain information in etymological structures.

The rest of the paper has been segregated as follows. Section 2 enlists the literature scrutinised for summarising this investigation, besides explaining the factors of software durability and quantum security alternatives, which have been explained. Furthermore, Section 2 also explains the multi-criteria decision-making, based symmetrical methodology of FAHP and FTOPSIS. Section 3 explains the quantitative analysis of the decision-making procedure for security durability in the quantum computing era, and Section 4 explains the sensitivity analysis of the results obtained. In addition, Section 3 explains the findings of the research in the discussion section. Section 4 concludes the study and enumerates the future research possibilities in determining SSD for the duration of the quantum computer.

2. Materials and Methods

2.1. Pertinent Works

A few exploration readings have been completed on quantum processing and programming security sturdiness. Unequivocally, the consolidated methodologies should be novel. Recently, specialists have zeroed in on the advancement of quantum computers and various calculations of quantum figures that improve the registering peculiarity. The quantum computer can, in a flash, settle the mind-boggling calculations of cryptography. The current balanced-based safety procedure, used by traditional computers and supercomputers, based on safety techniques, has been reduced in quantum processing time. Our exploration has especially highlighted the concerns and difficulties of programming security in the quantum period. In our research paper, we explicitly refer to the studies addressing this context.

In SSD, quantum-based security approaches are the most promising cryptographic field for faster, amazing, and more secure memos. Instead of the current prominence of estimations, subject to mathematical figuring advancement, quantum-based security component, and concealment, quantum possessions are of light under quantum mechanics in the cryptographic endeavor. Key movement licences inaccessible social affairs to make information-theoretical secure keys. The developers used an optical-based security methodology that will be equivalent to the qubit-based security approach [11–14]. The quantum key movement approach is intended to improve the security of the network.

The fuzzy AHP procedure provides assessment and weighting of the security factors in the selected software security [15]. This quantitative research identifies practical threats or security factors of a product, and the strategy of the fuzzy AHP decides the distinct weight of the components. The exploration study has additionally embraced the symmetrical technique based on fuzzy AHP, for assessment and examination of the effect on the security durability of software. Software security uses the multi-standards dynamic methodology of FAHP for the assessment of usable security in programming. Security is the essential portion of software [16]. The upgrading of security and ease of use in programming is of foremost significance. The developers have to guarantee the precise evaluation of the software. Security and durability are concepts that mention convenience, with security of the product being a trade-off between the two components: security and durability [17]. The developers selected the elements between ease-of-use and security, according to the necessity of the product. This entails that quantum key conveyance can appreciate the two sides of the world's common sense and security [18].

2.2. Software Durability and Quantum Security Technique

SSD are in the duration of quantum computing, characterized by the anticipated life span of software. Over the long haul, the efficient utilization of software in quantum computing requires security increments, on the grounds that can neutralize new security threats that keep evolving [19]. Creators have recognized and characterized the security solidness credits in their past work [20]. The different factors of security durability are usability, dependability, trustworthiness, and human trust, which can have utilized for the development of software security and durability in the quantum computing era. Durability factors and security alternatives are revealed in the Figure 1.

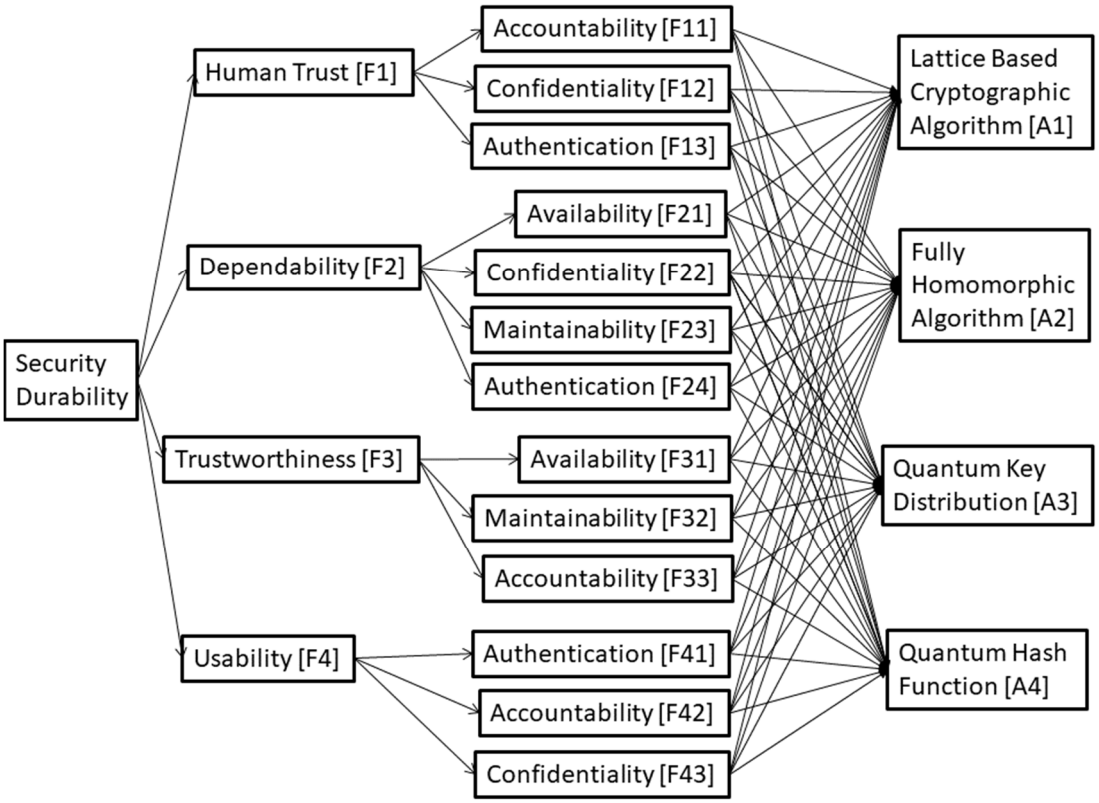

Figure 1. Schematic diagram of software durability factors and quantum security alternatives.

2.2.1. Human Trust

Compared to social cooperation, human belief is essentially characterized as a sensitive issue, where those who are believed upon have an ethical obligation to those confiding in them. In software and web application, the users' faith on the engineers is stated as human trust. The trust would imply that the product will work for normal length and secure information, promising effective SSD and, thus, optimum usability [21]. Security of the software and its durability in quantum computer existence will improve human trust and buyer's dependability [22]. Human trust is a readiness to depend on the product with certainty [23]. The asset of these variables is significant in building more grounded human trust. With the end goal of appraisal, traits of sturdy security as for human trust are at level 2.

2.2.2. Dependability

Software is secure if the client can rely upon it and its product to function as desired in the quantum computing era. The attributes of dependability, availability, maintainability confidentiality, and authentication assist in strengthening the security. This also influences the life expectancy of security administrations. Thus, it is directly identified with security credits, such as classification, validation, and unwavering quality. The other quantitative definition that determines whether the assistance is reliable or not is the capacity to evade administration disappointments that are more continuous and more serious than is satisfactory to the users [22]. The quantitative definition plans that steadfastness is likewise identified with accessibility and practicality. With the end goal of appraisal, characteristics of solid security and durability are regarding steadfastness at level 2.

2.2.3. Trustworthiness

The software possesses the trustworthiness in the developed period of quantum computing on the off chance that it proceeds as planned for a particular reason, when necessary, with new changes that have been done as of late, as well as without undesirable results,

practices, or credulous weaknesses. Trustworthiness is the affirmation that the software will proceed true to form [22]. There are numerous attributes of dependability, however, that have a couple of effects on software security in the quantum computing era. The attribute of trustworthiness is influencing the life expectancy of software security in the presence of the large quantum computer. Consequently, as indicated by its definition, trustworthiness relies upon availability, maintainability, and accountability. Further, security solidness necessitates that the product, in any event, works for a predetermined life span by fortifying the viability of security of software in the quantum era, thereafter refining the credibility of security. The quantitative characterization plans that credibility likewise identified with were availability, maintainability, and accountability. With the end goal of the evaluation, properties of tough security and durability, as for trustworthiness, were at level 2.

2.2.4. Usability

Software usability is a term to stay away from; however, it is utilized generally and there are recommendations that the term usability of software could be utilized in the case of all things being equal. The well-being of the basic software security and durability, in the quantum computing era, will be ensured by inducing quantum security algorithm or procedures in the software development life span [21], thus enhancing usability. For non-basic and less basic kinds of uses, the product usage measure is, notwithstanding, less obliged, which incites the nature of any product, bringing about lower durability of software, which can consequently instigate conduct in opposition to accessibility. Henceforth, the strength of these variables is significant in building more usable software. With the end goal of appraisal, traits of sturdy security, as for usability, were at level 2.

Additional, factors of software security, from the durability perspective, at level 3, are defined as:

- Availability: implies that the data is available for the approved clients as and when required. Availability, with regards a computer framework, alludes to the capacity of a client to get to data or assets for a predetermined term.
- Confidentiality: refers to allowing sanctioned admittance to susceptible and secure data.
- Authentication: the factor that answers for the personality of the client's profile. It is the interaction of deciding if a client is, indeed, who the client claims to be.
- Maintainability: the possibility that secure software will maintain or repair in the available environment or situation.
- Accountability: implies that each individual client who works with the product ought to have explicit duties regarding security confirmation. These errands incorporate singular obligation, as a component of the general security plan, since programming may become powerless by a dependable individual, such as a designer.

2.3. Quantum Algorithm

Quantum computers makes the balance, based on cryptography strategy, outdated. The quantum upgraded approach is the method of suspicion, wherein the quanta advances that are created cannot break the encryption calculation, such as AES, DES, Rjndael, etc. The present cryptographic calculations work effectively, yet the improvement of safety in quantum period would require more grounded processes. In the quantum improved circumstance, the key will get the cryptography cycle information. The quantum key appropriation is the innovative, well-adjusted approach of the organization encryption, where the engineer utilizes the vulnerability rule of the issue to guarantee that information cannot meddle with the software [22]. The quantum cryptography tactic manages diverse quantum dispersion keys, numerical centered methodology (for example, cross-section centered cryptographic methodology), hash centered mark, and code centered, which are helpful for security of programming and web application. The accompanying methodolo-

gies guarantee the product toughness. Quantum cryptography will guarantee the security of programming and, furthermore, guarantee the toughness of the product.

2.3.1. Quantum Key Distribution [A1]

Quantum key dissemination is the program of information, acknowledged as an encryption key with the assistance of qubits, which have remarkable conduct to traditional computer framework [22]. Until recently, the quantum key circulation required a different fiber optic technology-based line for the in-line move; yet, as of now, they can be moved from the presence fiber optic-based line. This diminishes the expense of correspondence. There is one more correspondence dependent on satellite correspondence. This method of correspondence depends on Einstein theory and is known as the 'spooky action at a distance' [23]. For the last couple of years, China has been dealing with the quantum correspondence protectorate. The correspondence rule is the trap; it is the interaction wherein the photons turn separately. At the point when we associate the turning of the photons, then, both have a connection. On the off chance that this connection is not sending the message, it implies that they can create an arbitrary number, which can be utilized in encryption calculation. This balanced technique of encryption is exorbitant.

2.3.2. Lattice-Based Cryptography Algorithm [A2]

In the environment of post quantum-based security procedures, this propositioned technique assures to protect the information counter to the quantum computing [24,25]. Hoffsten, Pipher, and Silverman brought together the lattice-based encryption, which is steady or unbreakable nowadays. The lattice-based erections are an n-dimensional intermittent gap, in which the n-dimensional vector $c1 \cdots \cdots cn \in R^n$ lattice-based generated set of vectors are shows in Equation (1).

$$\mathcal{L}(c1, \cdots, \cdots, \cdots, cn) = \left\{ \sum_{i=1}^{n} x_i c_i : \ x_i \epsilon \mathbb{Z} \right\}. \quad (1)$$

The vector $c1$, $c2$, $\cdots$, cn are recognized as elementary lattices [25]. Here, $\mathbb{Z}$ is the arbitrary session of lattice, R is the set of real numbers, and L is the dimension of lattice [26]. The problems of lattice-centered security procedures are the short vector problem (SVP) and Lenstra–Lenstra–Lovasz (LLL) algorithm. In SVP, participation lattice is indiscriminate, and its approximation is short. In 1982, the researcher gave the LLL algorithm. This algorithm has the approximation of $2^{O(n)}$, where n is the magnitude of lattice.

2.3.3. Fully Homomorphic Algorithm (FHA) [A3]

In FHA encryption, without uncovering the information, the information between two gatherings can be controlled by anybody, yet cannot be uncovered [27]. We can comprehend this idea by the case of a political decision technique. A political decision has two principle partners or gathering individuals, as well as the pioneers for whom individuals vote. In a political election, the election commission is the outsider in this setting, which can count the quantity of votes projected for every pioneer and uncover it before the allotted time. Subsequently, the information is in open area, yet be uncovered by the outsider, as because of FHA. The significant benefit of FHA encryption is that it cannot be broken in the post-quantum period. FHA encryption calculation arranges the public key to encode the information; while, for the decoding, we utilize the arithmetical capacity to address the encryption calculation and unscramble the calculation. The quantum computer cannot break the unscrambling system by arithmetical capacity.

2.3.4. Quantum Hash Function [A4]

Quantum hash function is defined as a hash function with any computational capacity that maps a subjective line of information to a fixed-length yield. As such, we need to pack any piece of information, i.e., names, federal retirement aide numbers, MP3 documents,

etc., into fixed-length esteems. For everything to fall into place, we need our hash capacities to work in a deterministic, public, and pseudorandom way. We need to realize that, for some random information 'x', the hash of x will consistently be the equivalent—that is, the hash is computationally controlled by its execution. A deterministic hash is futile, except if anybody can utilize it; so, we need the execution to be freely accessible to everybody. Likewise, we might want to darken the first contribution by guaranteeing the yield seems arbitrary. There are some major properties we can use to make hash function work. A hash function characterizes their properties, and it is the property that makes a hash function workable and helpful. We fundamentally need hash capacities to be single direction capacities. We need it to be not difficult to register the hash for x, yet we need it to be unrealistic, or incomprehensible, to invert the hash to discover x. We consider a hash function whose yield is 128-bit long. We would expect there to be 2^{128} potential yields—that is, more than 340 possibilities [28]. It seems that our 128-digit hash ought to be adequate for pretty much anything we toss at it.

2.4. Unified Technique of FAHP and FTOPSIS

In this research, we utilized the multi-model's dynamic, usual way of doing things for the assessment of elements, in regard to the strength of programming in the quantum time of safety. The crossover philosophy of FAHP was used to evaluate and assess the weight of the variables. The FTOPSIS gives the accurate positioning of the factors concerning the other current options. The fuzzy methodology of the accompanying philosophy was utilized for the accurate assessment of the product. We selected the quantum approach of safety as an option of safety. The FAHP and FTOPSIS approaches have the elements of assessment of the product. The dynamic issues are routinely utilized for meeting the clients' needs, as well as for the security strength of the programming. Numerous symmetrical techniques and assessment systems exist in writing and further the comprehension of the issues of programming security, in relation to the quantum period. In any case, for measuring the effectiveness of the safety of the product, FAHP is the most appropriate multi-rule approach. The neural network methodology, remembered for AHP, gives the most unstable size of dynamic. However, FAHP has a few additional challenges [29]. Hence, we incorporated the FTOPSIS and the dynamic way to deal with half-breed FAHP and FTOPSIS [30]. This is an exceptional technique, of sorts, that helps in the proficient evaluation of effect factors and its other options.

2.4.1. Fuzzy AHP Method

FAHP is the procedure of evaluating the efficient determination of the explicit issues in the security durability of software in a quantum computer. It depends on the characteristics and weights of the substitutes, essentially associated with those characteristics. The FAHP have philological standings, and their corresponding fuzzy numbers, signify the assessment processes. The philological standings have the following corresponding fuzzy numbers, as shown in Table 1.

Table 1. Fuzzy comparison measures or TFN.

Linguistic Terms	TFN
Equal	(1, 1, 1)
Not Bad	(2, 3, 4)
Good	(4, 5, 6)
Very Good	(6, 7, 8)
Perfect	(9, 9, 9)
Weak Advantage	(1, 2, 3)
Preferable	(3, 4, 5)
Fairly Good	(5, 6, 7)
Absolute	(7, 8, 9)

Afterward, FAHP procedure assesses every individual substance given by the analyst. The subsequent steps are determined using the triangular fuzzy number (TFN) from the hierarchal arrangement. The effect of the feature, and its option, is one measurement to determine various elective rules that have a pair-wise correlation of discrete factors, which assumes a crucial part in the order. The resulting step of FAHP changes the mathematical value from the linguistic terms by utilizing the fuzzy comparison measures [31]. FAHP procedure depends on deciding the weight of the components. The steps are given below.

Step 1: To drive the membership function from the triangular fuzzy number, which distributes the yes or no logic in many sub-values in Table 1, and $\prime\mu\prime$, as shown in Equation (2).

$$\mu_{\mathrm{a}}(\mathrm{x}) = \mathrm{a} \rightarrow [0, 1] \tag{2}$$

Let us select 'l' (lowest value), 'mi' (middle value), and 'u' (uppermost values), as shown in Figure 2.

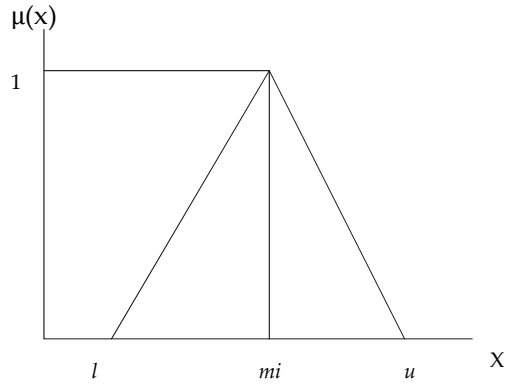

Figure 2. Triangular fuzzy number.

Step 2: Thereafter, we have evaluated the environment and transmuted the linguistic terms into TFN values. TFN estimation evaluates by the mathematical geometric mean comparison. The geometric mean was used to evaluate the significant result between factors.

Step 3: Further, we evaluated the two-dimensional analysis procedure of a fuzzy pair-wise comparison matrix (Equations (3) and (4)).

$$\widetilde{A^d} = \left[\widetilde{k}_{11}^d \widetilde{k}_{12}^d \ldots \widetilde{k}_{1n}^d,\ \widetilde{k}_{21}^d \widetilde{k}_{22}^d \ldots \widetilde{k}_{2n}^d, \cdots\cdots\cdots \widetilde{k}_{n1}^d \widetilde{k}_{n2}^d \ldots \widetilde{k}_{nn}^d\right] \tag{3}$$

$$\widetilde{k}_{ij} = \sum_{i=1}^{d} \widetilde{k}_{ij}^d \tag{4}$$

where, $\widetilde{k_{ij}^k}$ mentions the $'d'$ (decision-maker) on the condition i^{th} over j^{th} in Equations (2) and (3). If the preference is more than one, the average values are selected.

Step 4: The average preferences are evaluated, further creating the hierarchy of affecting factors. From Equation (5), the pair-wise appraisal matrixes are formed for all the affecting aspects in the ladder, basis of preferences.

$$\widetilde{A} = \lfloor \widetilde{k_{11}} \ldots \widetilde{k_{1n}} \cdots \ddots \cdots \widetilde{k_{n1}} \cdots \widetilde{k}_{nn} \rfloor \tag{5}$$

Step 5: The geometric mean and fuzzy weight of factors, derived by the Equation (6), which shows geometric mean technique; Equation (7), derives the fuzzy weight of factors.

$$\widetilde{p}_i = \left(\prod_{j=1}^{n} \widetilde{k}_{ij}\right)^{\frac{1}{n}},\ i = 1, 2, 3 \ldots\ldots n \tag{6}$$

$$\widetilde{w}_i = \widetilde{p}_i \otimes (\widetilde{p}_1 \oplus \widetilde{p}_2 \oplus \widetilde{p}_3 \ldots . \oplus \widetilde{p}_n)^{-1} \tag{7}$$

Step 6: Further, we derived and evaluated the normalized weight criteria from Equations (8) and (9).

$$M_i = \frac{\widetilde{w}_1 \oplus \widetilde{w}_2 \ldots \oplus \widetilde{w}_n}{n} \tag{8}$$

$$Nr_i = \frac{M_i}{M_1 \oplus M_2 \oplus \ldots \ldots \oplus M_n} \tag{9}$$

Step 7: The next step was to calculate the best non-fuzzy recital. The center of area methods is mentioned here which is the best non-fuzzy performance (BNP); the association and effect of the fuzzy weights of all metrics is calculated by Equation (10).

$$BNPwD1 = \frac{[(uw1 - lw1\ ) + (miw1 - lw1\ )]}{3} + lw1 \tag{10}$$

2.4.2. Fuzzy TOPSIS Method

The 'M' option in the mathematical geometrical mean plan, with the 'M' point and 'N' dimensional region TOPSIS MCDM approach, is consumed in multi-measure choices for positioning. The TOPSIS strategies is principally found on the possibility of the absolute and furthest separation from the positive ideal planning, the adverse perfect answer for ideal and least ideal arrangements, individually [32]. The TOPSIS approach is valuable for apportioning the ideal situation of the other option and factor, regarding the rules. To accomplish consistency with the fuzzy climate, TOPSIS relegates the fuzzy number, as indicated by the inclination and addresses the meaning of models. We selected the hybrid approach of FAHP-FTOPSIS, in order to facilitate the collective choice decision-making approach in a fuzzy climate. FTOPSIS techniques have the following steps.

Step 1: Further estimation of the ranks of the factors, by FTOPSIS and FAHP approaches, is used to evaluate the weight of the factors with the selected alternatives, as mentioned above in Figure 1.

Step 2: In FTOPSIS, firstly, derive the table for the linguistic terms, used in the affecting factors and alternatives from Table 2, as mentioned below. Further, we used the fuzzy decision matrix, with the assistance of Equation (11) and evaluated the matrix.

$$\begin{array}{cc} C_1 \quad \ldots\ldots & C_n \\ \widetilde{K} = \begin{array}{c} A_1 \\ \ldots \\ A_m \end{array} & \begin{bmatrix} \widetilde{x}_{11} & \cdots & \widetilde{x}_{1n} \\ \ldots & \ddots & \ldots \\ \widetilde{x}_{m1} & \cdots & \widetilde{x}_{mn} \end{bmatrix} \end{array} \tag{11}$$

Table 2. Linguistic terms and its TFN.

Variable	TFN
Very scanty	(0, 1, 3)
Scanty	(1, 3, 5)
Light	(3, 5, 7)
Satisfactory	(5, 7, 9)
Very satisfactory	(7, 9,10)

Here, $\widetilde{x}_{ij} = \frac{1}{D}\left(\widetilde{x}_{ij}^1 \cdots \oplus \widetilde{x}_{ij}^d \oplus \cdots \widetilde{x}_{ij}^D\right)$, $\widetilde{x}_{ij}^d$, is used to calculate the ranking of the alternative A_i, the factor C_J *is* evaluated by the d^{th} practitioner $\widetilde{x}_{ij}^d = (l_{ij}^d, mi_{ij}^d, u_{ij}^d)$.

Step 3: The normalized fuzzy decision matrices, evaluated by Equation (12) and represented by $\widetilde{P}$. The normalization is calculated by the Equation (13).

$$\widetilde{P} = \left[\widetilde{p}_{ij}\right]_{m \times n} \tag{12}$$

$$\widetilde{p}_{ij} = \left(\frac{l_{ij}}{u_j^+}, \frac{mi_{ij}}{u_j^+}, \frac{u_{ij}}{u_j^+}\right), u_j^+ = max\{u_{ij},\ i = 1,2,3\ldots n\} \tag{13}$$

The most expected level (u_j^+) is 1, and the worst is 0. The normalization process TFNs are calculated by a similar step.

Step 4: Further, the weighted normalized fuzzy decision matrix ($\widetilde{Q}$) is quantified by Equation (14).

$$\widetilde{Q} = \left[\widetilde{q}_{ij}\right]_{m\times n} i = 1,2,\ldots m;\ j = 1,2,3\ldots n \tag{14}$$

where, $\widetilde{q}_{ij} = \widetilde{p}_{ij} \otimes \widetilde{w}_{ij}$.

Step 5: The fuzzy positive ideal clarification 'A^+' and fuzzy negative ideal clarification 'A^-' are calculated; the best and worst solutions, respectively, were Equations (15) and (16). This can be done by avoiding the irregular complication of calculation.

$$A^+ = \left(\widetilde{q}_{1,\cdots\ldots}^*\widetilde{q}_{j,\cdots\ldots}^*\widetilde{q}_{n,}^*\right) \tag{15}$$

$$A^- = \left(\widetilde{q}_{1,\cdots\ldots}^*\widetilde{q}_{j,\cdots\ldots}^*\widetilde{q}_{n,}^*\right) \tag{16}$$

The detachments of alternative were calculating with Equations (17) and (18), respectively.

$$\widetilde{d}_i^+ = \sum_{j=1}^{n} d\left(\widetilde{q}_{ij}, \widetilde{q}_{ij}^*\right), i = 1,2,\ldots m;\ j = 1,2,3\ldots n \tag{17}$$

$$\widetilde{d}_i^- = \sum_{j=1}^{n} d\left(\widetilde{q}_{ij}, \widetilde{q}_{ij}^*\right), i = 1,2,\ldots m;\ j = 1,2,3\ldots n \tag{18}$$

Step 6: Further, the closeness coefficient, represented by $C\widetilde{C}_i$, is defined as the relative close degree of the alternatives, used in the security and durability of software. It has been evaluated here with Equation (19). The closeness coefficients determine the desired levels of closeness. The closeness coefficients evaluate the fuzzy gaps level at the origin of fuzzy closeness to recover the alternatives [33]. The detachments of the best and the worst levels of alternatives have been calculated.

$$C\widetilde{C}_i = \frac{\widetilde{k}_i^-}{\widetilde{k}_i^+ + \widetilde{k}_i^-} = 1 - \frac{\widetilde{k}_i^+}{\widetilde{k}_i^+ + k_i^-},\ i = 1,2,\ldots.,m \tag{19}$$

The ranks of the alternatives were determined by Equation (19) by using the detachments. Further, the calculations of the security and durability of software, in the era of quantum computing, with the help of factors and selected alternatives, are done, and the numerical analysis are explained in next section of the paper.

3. Numerical Data Analysis

The symmetrical approach of FAHP gives the weight of factors in level 1, further classified in level 2, as presented in Figure 1. The approach FAHP assesses the factors' weight, with respect to the alternatives. The procedure FTOPSIS measure gives the ranking of security durability alternatives. Based on the weights and ranks of the security durability factors and their alternatives, we decided the level of closeness and examined whether this outcome ought to be utilized for developing the software, in the period of the quantum computing era. As presented in Figure 1, the characteristic of the request at one level was influenced by another; however, its impact was not identical on them. It may change. With the ultimate objective of evaluation, we changed the assembled assets into manacles of significance and showed it in Figure 2. For the confirmation of valuation, factors of mystery with admiration to sensible security at level 2 are addressed as F11, F12, ... ; properties of uprightness concerning functional security at level 2 are addressed as F11, F12, ... ; as portrayed in Figure 1, with the help of these features of significance, we surveyed the

SSD in the quantum computing duration. For social affairs, the data, with the help of Equations (2)–(19), security of software, through hybrid FAHP and FTOPSIS, have been evaluated as follows.

With the help of Table 1 and Equations (2)–(19), the etymological characteristics were changed into numeric characteristics, and TFNs were determined esteems. The calculations of the assessment, along with the TFNs characteristics, are enlisted from Tables 3–7.

Table 3. Subjective cognition results of evaluators in linguistic terms.

	A1	A2	A3	A4
F11	2.4500, 4.2700, 6.2700	1.3600, 3.3600, 5.3600	0.6400, 2.2700, 4.2700	2.4500, 4.2700, 6.2700
F12	4.6400, 6.6400, 8.5500	0.8200, 2.6400, 4.6400	5.3600, 7.3006, 8.7300	4.6400, 6.6400, 8.5500
F13	5.3600, 7.3006, 8.7300	5.5500, 7.5500, 8.9100	2.4500, 4.2700, 6.2700	5.3600, 7.3006, 8.7300
F21	3.7300, 5.5500, 7.2700	4.4500, 6.4500, 8.1800	4.6400, 6.6400, 8.5500	3.7300, 5.5500, 7.2700
F22	2.3600, 4.2700, 6.2700	2.4500, 4.2700, 6.2700	5.3600, 7.3006, 8.7300	2.3600, 4.2700, 6.2700
F23	5.3600, 7.3006, 8.7300	5.5500, 7.5500, 8.9100	3.7300, 5.5500, 7.2700	5.3600, 7.3006, 8.7300
F24	3.7300, 5.5500, 7.2700	4.4500, 6.4500, 8.1800	2.3600, 4.2700, 6.2700	3.7300, 5.5500, 7.2700
F31	2.4500, 4.2700, 6.2700	1.3600, 3.3600, 5.3600	5.3600, 7.3006, 8.7300	5.5500, 7.5500, 8.9100
F32	4.6400, 6.6400, 8.5500	0.8200, 2.6400, 4.6400	3.7300, 5.5500, 7.2700	4.4500, 6.4500, 8.1800
F33	5.3600, 7.3006, 8.7300	5.5500, 7.5500, 8.9100	1.6400, 3.5500, 5.5500	3.7300, 5.5500, 7.2700
F41	3.7300, 5.5500, 7.2700	4.4500, 6.4500, 8.1800	1.3600, 3.3600, 5.3600	2.3600, 4.2700, 6.2700
F42	2.3600, 4.2700, 6.2700	2.4500, 4.2700, 6.2700	0.8200, 2.6400, 4.6400	4.8200, 6.8200, 8.5500
F43	5.3600, 7.3006, 8.7300	5.5500, 7.5500, 8.9100	5.3600, 7.3600, 8.7300	1.4500, 3.3600, 5.3600

Table 4. Normalized fuzzy decision matrix.

	A1	A2	A3	A4
F11	0.5900, 0.8000, 0.9600	0.4600, 0.6800, 0.8800	0.5900, 0.8000, 0.9600	0.4600, 0.6800, 0.8800
F12	0.5400, 0.7500, 0.9200	0.5200, 0.7400, 0.9300	0.5400, 0.7500, 0.9200	0.5400, 0.7500, 0.9200
F13	0.5400, 0.7500, 0.9200	0.5200, 0.7400, 0.9200	0.5400, 0.7500, 0.9200	0.5400, 0.7500, 0.9200
F21	0.5900, 0.8000, 0.9600	0.4600, 0.6800, 0.8800	0.3500, 0.5800, 0.8100	0.3500, 0.5800, 0.8100
F22	0.5400, 0.7500, 0.9200	0.5200, 0.7400, 0.9300	0.4600, 0.6700, 0.8600	0.4600, 0.6700, 0.8600
F23	0.5400, 0.7500, 0.9200	0.5400, 0.7500, 0.9200	0.5200, 0.7400, 0.9300	0.5000, 0.7100, 0.8900
F24	0.3500, 0.5800, 0.8100	0.5400, 0.7500, 0.9200	0.5200, 0.7400, 0.9200	0.4600, 0.6700, 0.8600
F31	0.4600, 0.6700, 0.8600	0.3500, 0.5800, 0.8100	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900
F32	0.5000, 0.7100, 0.8900	0.4600, 0.6700, 0.8600	0.5200, 0.7400, 0.9200	0.5400, 0.7500, 0.9200
F33	0.4600, 0.6700, 0.8600	0.5000, 0.7100, 0.8900	0.4600, 0.6800, 0.8800	0.3500, 0.5800, 0.8100
F41	0.5000, 0.7100, 0.8900	0.4600, 0.6700, 0.8600	0.5200, 0.7400, 0.9300	0.4600, 0.6700, 0.8600
F42	0.5000, 0.7100, 0.8900	0.5000, 0.7100, 0.8900	0.5400, 0.7500, 0.9200	0.5200, 0.7400, 0.9300
F43	0.4600, 0.6700, 0.8600	0.3800, 0.6000, 0.8000	0.5400, 0.7500, 0.9200	0.5200, 0.7400, 0.9300

Table 5. Weighted normalized fuzzy decision matrix.

	A1	A2	A3	A4
F11	0.00200, 0.00600, 0.02000	0.00300, 0.01200, 0.04200	0.00200, 0.00900, 0.03000	0.00200, 0.01000, 0.03500
F12	0.00200, 0.00800, 0.02500	0.00200, 0.00600, 0.02000	0.00200, 0.00600, 0.02000	0.00300, 0.01200, 0.04200
F13	0.00200, 0.00700, 0.02200	0.00200, 0.00800, 0.02500	0.00200, 0.00800, 0.02500	0.00200, 0.00600, 0.02000
F21	0.00200, 0.00600, 0.02000	0.00200, 0.00700, 0.02200	0.00200, 0.00700, 0.02200	0.00200, 0.00800, 0.02500
F22	0.00200, 0.00800, 0.02500	0.00200, 0.00600, 0.02000	0.00200, 0.00600, 0.02000	0.00200, 0.00700, 0.02200
F23	0.00300, 0.01200, 0.04100	0.00200, 0.00800, 0.02500	0.00200, 0.00800, 0.02500	0.00200, 0.00600, 0.02000
F24	0.00300, 0.01200, 0.04200	0.00200, 0.00700, 0.02200	0.00300, 0.01200, 0.04100	0.00200, 0.00800, 0.02500
F31	0.00200, 0.00600, 0.02000	0.00200, 0.00600, 0.02000	0.00300, 0.01200, 0.04200	0.00300, 0.01200, 0.04100
F32	0.00300, 0.01200, 0.04100	0.00200, 0.00800, 0.02500	0.00200, 0.00600, 0.02000	0.00300, 0.01200, 0.04200
F33	0.00300, 0.01200, 0.04200	0.00300, 0.01200, 0.04100	0.00200, 0.00600, 0.02000	0.00200, 0.00600, 0.02000
F41	0.00200, 0.00600, 0.02000	0.00300, 0.01200, 0.04200	0.00200, 0.00800, 0.02500	0.00200, 0.00600, 0.02000
F42	0.00200, 0.00800, 0.02500	0.00200, 0.00600, 0.02000	0.00200, 0.00700, 0.02200	0.00200, 0.00800, 0.02500
F43	0.00200, 0.00700, 0.02200	0.00200, 0.00700, 0.02200	0.00200, 0.00700, 0.02200	0.00200, 0.00700, 0.02200

Table 6. Closeness coefficient of the detachments level among alternatives.

Alternatives		d+i	d-i	Gap Degree of CC+i	Satisfaction Degree of CC-i
Alternative 1	A1	0.0548547	0.03685647	0.365886957	0.625232141
Alternative 2	A2	0.0648599	0.03556857	0.524658547	0.644223521
Alternative 3	A3	0.0488574	0.05455658	0.569775847	0.444112547
Alternative 4	A4	0.0496587	0.03688574	0.256112365	0.527001245

Table 7. Sensitivity analysis.

Experiments	Weights/Alternatives		A1	A2	A3	A4
Experiment-0	Original Weights		0.625232141	0.644223521	0.444112547	0.527001245
Experiment-1	F11		0.664114542	0.542556587	0.489455487	0.672775847
Experiment-2	F12		0.659558471	0.534525471	0.499556587	0.607125524
Experiment-3	F13		0.549554874	0.407635257	0.396885471	0.521223254
Experiment-4	F21		0.549885674	0.407658254	0.395565547	0.593556587
Experiment-5	F22		0.659556547	0.535226535	0.498554745	0.607652511
Experiment-6	F23	Satisfaction Degree (CC-i)	0.581112547	0.407965587	0.396122011	0.485556571
Experiment-7	F24		0.549225635	0.431563547	0.383526587	0.607652113
Experiment-8	F31		0.549885684	0.728854474	0.541225474	0.521001245
Experiment-9	F32		0.656525471	0.535556587	0.499565241	0.607235264
Experiment-10	F33		0.581225358	0.445223525	0.427002154	0.522265254
Experiment-11	F41		0.5495568574	0.408547444	0.396322154	0.485885474
Experiment-12	F42		0.6598854741	0.431556587	0.383565225	0.607852145
Experiment-13	F43		0.5812235654	0.728855564	0.540000154	0.520025214

Sensitivity Analysis

The sensitivity analysis is determined for the closeness of every factor. The sensitivity analysis was calculated to reduce the complexity of the outcome values. The sensitivity analysis [34] is obtainable in Table 7. The sensitivity analyses calculate the precise weight of the factors in the security durability of the software in the quantum computing era. The sensitivity analysis is confirmed by numerous analyses of each factor; the various trials show the various outcomes, as seen in Table 7; its graph is shown in Figure 3. Further, from the sensitivity analysis, the closeness coefficients were determined, and the satisfaction degree was evaluated. The calculated weight of every factor (F1 to F4 taken as a consistent at level 1, with further sublevels having factors in level 2) was determined, and by the hybrid procedure of FAHP and FTOPSIS, we evaluated the satisfaction degree.

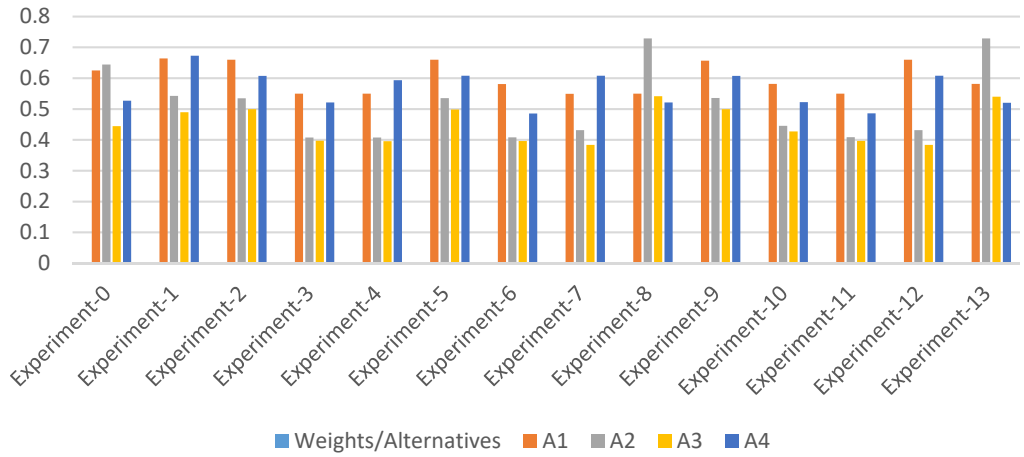

Figure 3. Graphical representation of sensitivity analysis.

4. Discussion

The hybrid decision-making approach for software security and durability, in the presence of quantum computers and lattice-based cryptography alternative procedures for security, are considered the most significant procedures for software durability in the

era of quantum computing. Hence, the necessity for this period is to change to viable and software synthesis, in the time of quantum computing. This assessment is focused on the two and has a different, evened-out structure, which determines the contributory factors in the acceptable security plan of the product. The assessment of the efficacy of safety programming is the ideal approach to achieve doable toughness. This paper investigates security durability, similarly to practicality factors, and evaluates sensible and reliable security, with regard to the arrangement. The outcomes of the assessment will help the specialists incorporate legitimate security durability into software for the full-fledged advancement of quantum computers.

An expansive system of exploration refers to the turn of events and assessment of the safety and durability of programming, in the time of the quantum computer. This paper specifies the solidity factors at two levels, thus providing more security choices, as per quantum security. Our examination procedure will assist the designer or architect in building the product with appropriate improvements in the advancement life pattern of software. There are not many assessment models or methodologies available in the composition for assessing security solidness freely, i.e., the openness of models or procedures that facilitate security on the fuzzy AHP system inside and out. In this work, we have made four quantum security choices for the product plan.

Revelations from this work can be used to assess the security durability of software and will assist the designers in achieving security alleviation and other related designs for significant security issues, accordingly, providing secure and sturdy programming in the time of quantum computing.

The quantitative consequences, accomplished by FAHP and FTOPSIS, will provide the experts with the means to order the higher-positioned parts of a product in the board structure.

The FAHP strategy gives the weightiness of the SSD factors; FTOPSIS gives the position or rank of the accompanying alternatives in the quantum computing era.

The SSD, in the quantum period, should be the preeminent need for both future examinations and present undertakings to enhance the adequacy of software. This assessment would help the developers acquire information about the design of software and web applications and their security and durability.

Enhancement instructions, at the end of this evaluation, will help the specialists in decontaminating the construction of safety using high coordinated angles in apprehension. This assessment may have a couple of delimitations, which can be addressed in future examinations. The delineating outcomes are: the data assembled for secure and sturdy programming improvement is important; the consequences may differ if the data is gigantic; and there may be additional security arrangements involving people who are not included in this work.

5. Conclusions

Quantum-based applications are currently being created across the globe. Various emerging issues, in the context of quantum registering, require intensive research. This paper explored some of the essential parts of quantum computing and, additionally, researched the ability of quantum handling to work on the logical and figure limits, while handling programming security and solidity. Clearly, creating a totally protected system is beyond the domain of the creative mind. Therefore, the objective of surveying SSD does not assure the incredible feat of absolutely secure programming. The objective here is to find mechanisms that can ensure elongated time-serviceable programming. Thus, a dedicated focus on reinforcing the product's durability, from the earliest starting point of the advancement sequence, will determine the level of significant worth of the product. In this research article, we perused the quantum-based security approach and SSD for the duration of the quantum computer. The calculation of quantum key dissemination will reveal the quantum-based security threat to the current encryption security. Regardless, whether it is lattice-based quantum computing or not, there can be no guarantee of fool-

proof security for online communication. However, given the advancement of quantum computing, lattice-based quantum computing is definitely a powerful improvisation.

Author Contributions: All the authors have contributed equally to the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: The project has been funded by Taif University, Kingdom of Saudi Arabia.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to government data policy.

Acknowledgments: This research was supported by Taif University Researchers Supporting Project number (TURSP-2020/306), Taif University, Taif, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Vijayan, J. Application Security Risk Report 2019: 6 Takeaways for Your Team, TechBeacon. Available online: https://techbeacon.com/security/application-security-risk-report-2019-6-takeaways-your-app-sec-team (accessed on 19 November 2021).
2. Alashaikh, A.; Tipper, D.; Gomes, T. Embedded network design to support availability differentiation. *Ann. Telecommun.* **2019**, *74*, 605–623. [CrossRef]
3. Song, L.; Zhang, J.; Mukherjee, B. Dynamic provisioning with availability guarantee for differentiated services in survivable mesh networks. *IEEE J. Sel. Areas Commun.* **2007**, *25*, 35–43. [CrossRef]
4. Alenezi, M.; Kumar, R.; Agrawal, A.; Khan, R.A. Usable-security attribute evaluation using fuzzy analytic hierarchy process. *ICIC Express Lett.* **2019**, *13*, 453–460.
5. Agrawal, A.; Alenezi, M.; Khan, S.A.; Kumar, R.; Khan, R.A. Multi-level Fuzzy system for usable-security assessment. *J. King Saud Univ.-Comput. Inf. Sci.* **2019**. [CrossRef]
6. Kumar, R.; Khan, A.I.; Abushark, Y.B.; Alam, M.M.; Agrawal, A.; Khan, R.A. An Integrated Approach of Fuzzy Logic, AHP and TOPSIS for Estimating Usable-Security of Web Applications. *IEEE Access* **2020**, *8*, 50944–50957. [CrossRef]
7. Kelty, C.; Erickson, S. *The Durability of Software*; Meson Press: Lüneburg, Germany, 2015; pp. 1–13.
8. Nathan, E. When Good Software Goes Bad: The Surprising Durability of an Ephemeral Technology. In Proceedings of the MICE (Mistakes, Ignorance, Contingency, and Error) Conference, Munich, Germany, 2–4 October 2014; pp. 1–16.
9. Firesmith, D.G. *Common Concepts Underlying Safety. Security and Survivability Engineering*; Technical Note CMU/SEI2003-TN033; Software Engineering Institute: Pittsburg, PA, USA, 2003; Volume 1, pp. 1–75.
10. Becker, S.; Boskovic, M.; Dhama, A. Trustworthy Software Systems: A Discussion of Basic Concepts and Terminology. *ACM SIGSOFT Softw. Eng. Notes* **2006**, *31*, 1–18. [CrossRef]
11. Arute, F.; Arya, K.; Babbush, R. Quantum supremacy using a programmable superconducting processor. *Nature* **2019**, *574*, 505–510. [CrossRef] [PubMed]
12. Shor, P.W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM J. Comput.* **1997**, *26*, 1484–1509. [CrossRef]
13. Mitra, S.; Jana, B.; Bhattacharya, S.; Pal, P.; Poray, J. Quantum cryptography: Overview, security issues and future challenges. In Proceedings of the 4th International Conference on Opto-Electronics and Applied Optics (Optronix), Kolkata, India, 2–3 November 2017; pp. 1–7.
14. Abomhara, M.; Køien, G.M. Cyber security and the internet of things: Vulnerabilities, threats, intruders and attacks. *J. Cyber Secur. Mobil.* **2015**, *1*, 65–88. [CrossRef]
15. Ma, X.; Zeng, P.; Zhou, H. Phase-matching quantum key distribution. *Phys. Rev. X* **2018**, *8*, 031043. [CrossRef]
16. Zhang, W.; Ding, D.S.; Sheng, Y.B.; Zhou, L.; Shi, S.B.; Guo, G.C. Quantum secure direct communication with quantum memory. *Phys. Rev. Lett.* **2017**, *118*, 220501. [CrossRef] [PubMed]
17. Pang, X.L.; Qiao, L.F.; Sun, K.; Liu, Y.; Yang, A.L.; Jin, X.M. Experimental Quantum-enhanced Cryptographic Remote Control. *Sci. Rep.* **2019**, *9*, 5809. [CrossRef] [PubMed]
18. Ajtai, M. Generating hard instances of lattice problems. In *Complexity of Computations and Proofs, Quad*; Weizmann Institute of Science: Rehovot, Israel, 2004; Volume 16, pp. 1–32.
19. Sen, J. *Homomorphic Encryption—Theory and Practice of Cryptography and Network Security Protocols and Technologies*; IntechOpen: London, UK, 2013; Volume 1, pp. 1–10.
20. Bernstein, D.J. *Introduction to Post-Quantum Cryptography*; Springer: Berlin/Heidelberg, Germany, 2009; Volume 1, pp. 1–10.
21. Alzahrani, F.A.; Ahmad, M.; Nadeem, M.; Kumar, R.; Khan, R.A. Integrity assessment of medical devices for improving hospital services. *Comput. Mater. Contin.* **2021**, *67*, 3619–3633. [CrossRef]
22. Ladd, T.; Jelezko, F.; Laflamme, R. Quantum computers. *Nature* **2010**, *464*, 45–53. [CrossRef] [PubMed]

23. Bernstein, D.; Lange, T. Post-quantum cryptography. *Nature* **2017**, *549*, 188–194. [CrossRef] [PubMed]
24. Hoffstein, J.; Pipher, J.; Silverman, J.H. Lattices and Cryptography. In *An Introduction to Mathematical Cryptography. Undergraduate Texts in Mathematics*; Springer: New York, NY, USA, 2014.
25. Micciancio, D.; Regev, O. *Lattice-Based Cryptography*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 1–20.
26. Howe, J.; Khalid, A.; Rafferty, C.; Regazzoni, F.; O'Neill, M. On Practical Discrete Gaussian Samplers for Lattice-Based Cryptography. *IEEE Trans. Comput.* **2018**, *67*, 322–334. [CrossRef]
27. Zech, P.; Felderer, M.; Breu, R. Towards Risk—Driven Security Testing of Service Centric Systems. In Proceedings of the 2012 12th International Conference on Quality Software, Xi'an, China, 27–29 August 2012; pp. 140–143. [CrossRef]
28. Alenezi, M.; Nadeem, M.; Agrawal, A.; Kumar, R.; Khan, R.A. Fuzzy multi criteria decision analysis method for assessing security design tactics for web applications. *Int. J. Intell. Eng. Syst.* **2020**, *13*, 181–196. [CrossRef]
29. Memari, A.; Dargi, A.; Jokar, M.R.A.; Ahmad, R.; Rahim, A.R.A. Sustainable supplier selection: A multi-criteria intuitionistic fuzzy TOPSIS method. *J. Manuf. Syst.* **2019**, *50*, 9–24. [CrossRef]
30. Solanki, R.; Gulati, G.; Tiwari, A.; Lohani, Q.M.D. A correlation based Intuitionistic fuzzy TOPSIS method on supplier selection problem. In Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Vancouver, BC, Canada, 24–29 July 2016; pp. 2106–2112.
31. Jalali, M.S.; Razak, S.; Gordon, W.; Perakslis, E.; Madnick, S. Health care and cybersecurity: Bibliometric analysis of the literature. *J. Med Internet Res.* **2019**, *21*, e12644. [CrossRef] [PubMed]
32. Pujolle, G.; Serhrouchni, A.; Ayadi, I. Secure session management with cookies. In Proceedings of the 7th International Conference on Information, Communications and Signal Processing (ICICS), Macau, China, 8–10 December 2009; pp. 1–6.
33. Li, Q. An Improved fuzzy AHP approach to evaluating conductor joint alternatives. In Proceedings of the Seventh International Conference on Fuzzy Systems and Knowledge Discovery, Yantai, China, 10–12 August 2010; pp. 811–814.
34. Öztaysi, B.; Onar, S.Ç.; Boltürk, E.; Kahraman, C. Hesitant fuzzy analytic hierarchy process. In Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Istanbul, Turkey, 2–5 August 2015; pp. 1–7.