




Dissecting the Security and Usability Alignment in the Industry

Bilal Naqvi^(✉) 

Software Engineering, LENS, LUT University, 53850 Lappeenranta, Finland
Syed.naqvi@lut.fi

Abstract. Security and usability are two important characteristics often in conflict with each other. This paper presents challenges related to alignment between security and usability in the industry. The challenges were identified after analyzing the data from 12 semi-structured interviews. There were nine different challenges in industrial practices which were identified after the interviews, moreover, two recommendations for future solutions were also identified. The paper also presents a framework for addressing the identified challenges within the industry context. The framework presented in the paper has been tailored for the agile development context and aims at identifying minimal trade-offs between security and usability.

Keywords: Usability · Security · Usable security · Framework · Trade-offs

1 Introduction

The human facet of security more commonly referred to as usable security aims at bridging the aspects of usability (effectiveness, efficiency, and satisfaction while using the system [3]) and principles of security (confidentiality, integrity, and availability, among others [19]) in the development of software systems. Despite the realization that security systems should be usable, humans are often blamed as the weakest link in the security chain. Research on human psychology identifies that all the mistakes people commit are predictable. Either these mistakes occur due to latent failures (organizational conditions and practices) or due to active failures (individual human factors) [1]. The factors leading to latent failures include productivity-driven environments, lack of training, interruption in tasks, poor equipment, etc. However, the active failures which occur due to human errors are also influenced by the organizational conditions in addition to individual human factors such as risk-taking attitudes, inexperience, limitations of memory, etc. One example in this regard is the successful cyber-attack on Victorian regional hospitals in Australia, where the need for effective usable security was realized as the human facet of security was compromised leading to a ransomware attack [18]. The attack affected all hospital systems including patient records, booking, and management systems, doctors were not able to access patients' health records either. A combination of latent and active failures led to a successful attack.

One important aspect which contributes to most security failures is the lack of user-centered design of security mechanisms [2]. The development approach has been focused on fixing the humans to be able to use the system, rather than designing the systems duly considering appropriate mental models and user perceptions about security [1]. However, there is a realization that it is vital to consider the aspects of usability in the security design as a key factor of security hygiene. Otherwise, the developed systems and services despite being secure against external threats could be susceptible to user mistakes leading to a security failure.

The latent failures are induced due to malpractices in the organizational conditions and practices, however, there is also an impact of these organizational practices in determining the active failures. To elaborate on this aspect this paper reports findings from semi-structured interviews conducted with front-end developers, user experience experts, security engineers, and product owners working in a leading European IT organization. The paper reports the gaps in organizational practices (latent failures) which lead to the development of complex secure systems thereby making the systems susceptible to active failures. During the interviews with experts, it was intended to identify the importance of security as a product quality characteristic and that of usability both as product quality and as a quality characteristic in use (*usability in use*) [3]. It was also intended to identify how security and usability issues specifically conflicts between the two are aligned during the system development life cycle, the intent was to identify best practices and mechanisms for handling the conflicts from the industry.

Furthermore, based on the findings of the interviews the paper presents a framework for improving the current state of the art. The framework is an adapted version of the framework presented in [4], however, a significant difference is that the current framework is applicable for agile development contexts. The initial version of the framework considering the challenges identified after the interviews was subject to validation by involving the interviewees in a workshop where the interview findings and the framework were presented. However, after incorporating the comments, the framework was updated which is also presented in this paper.

The remainder of the paper is organized as follows. Section 2 presents the background. Section 3 presents the interview protocol and results. Section 4 presents the framework, and Sect. 5 concludes the paper.

2 Background

Before presenting the challenges related to alignment between security and usability in the industry, one additional challenge related to alignment between security and usability was identified after analyzing the existing literature on the topic. Different communities and interest groups including usable security community, traditional computer security community, human-computer interaction (HCI) community, and software engineering community have been studying the relationships between security and usability. The study of security and usability dependencies by different communities and interest groups from their respective viewpoints has led to inconsistent perceptions [5].

2.1 Trade-offs

Most of the work on security and usability dependencies advocates the existence of trade-offs [6–10]. A case study on iOS and Android was conducted to find an answer for “what is more important: usability or security” [6]. The results identify that the importance of security and usability is purely situation-based and that the trade-offs are sometimes in favor of security and vice versa. Furthermore, based on the comparison of the two platforms, the study identified that android takes the lead in usability as compared to iOS. However, security is a preferred feature in iOS devices.

Concerning the dimensions of the conflict discussed earlier, sometimes the trade-offs are in favor of security and vice versa. From the usability of security dimension, password masking is implemented in most of the authentication mechanisms to protect against shoulder surfing, but at the cost of the usability element of ‘feedback.’ Other conflicts leading to trade-off situations may arise when critical security decision-making is reliant on the users. Security developers do not consider the fact that the users are less knowledgeable than the implementers, and that the users should be presented with high-level yet comprehensive information. The trade-off between security and usability is because security is considered a burden both by the developers and by the system users [7].

From the security of usability dimension, the location awareness capability of smartphones remains enabled until disabled manually. This is done to ensure UX in applications like maps, weather updates, options near me, etc. This comes at the cost of privacy and has security implications as well since the users’ location data can be subjected to unauthorized disclosure using one of the prevalent mechanisms.

Irrespective of the type of system under consideration, there is evidence of the existence of trade-offs between security and usability [8, 9], for instance, security and usability trade-offs in end-to-end email encryption. The results of the study [8] identified that the participants in their choice of the preferred system deliberately made trade-offs between security and usability. Another case study [10] for handling security and usability in database systems identifies that the systems designed with tight security have limited usability. In other words, robust security comes at the cost of usability. Therefore, it is a trade-off versus usability.

Furthermore, researchers extend the argument of trade-offs to propose that quantification of trade-offs can contribute to achieving an effective balance between security and usability [9]. Therefore, a study was conducted to test and quantify possible usability and security trade-offs using three different schemes for e-voting systems [11]. The results reveal that the voters were in favor of more secure systems and were willing to sacrifice a maximum of 26 points (scale of 0 to 100) on usability for a system that provides higher security. The authors state, “nevertheless, the security gains come at the cost of usability losses”.

2.2 No Trade-offs

In parallel to the research identifying the existence of trade-offs, some researchers classify usability and security trade-offs as mere myths, and that security and usability are not inherently in conflict.

A special issue ‘the security-usability trade-off myth’ features a discussion of researchers and practitioners in usable security [12]. The participants were of the view that decreasing usability can lead to less security. The participants discussed the example of two-factor authentication involving a one-time password (OTP) and its consequences if the length of OTP is increased from 6 to 8 characters, which represents the case of a false trade-off. There are cases where increased usability can lead to increased security, for example making the security functionality more understandable can lead to improved user decision-making and increased security. Overall, the participants were of the view that “security experts simply invoke the myth of trade-off between usability and security and use this as a cover to avoid the exercise of saying precisely what security benefit in precisely what scenarios this usability burden is going to deliver”.

As a step further from the argument of no trade-offs, there is a need to incorporate the aspects of user value-centered design [13]. A framework to identify user values associated with security systems and services is required. There is a need for shifting the approach of fixing the users to be ‘able to use’ security. Therefore, incorporating value-sensitive design, which can help, requires the following actions, (1) identify and document user behavior drivers, trends, and patterns, which might conflict with security mechanisms. (2) conduct value-sensitive conceptual and empirical analyses for the security application. The authors state “identifying the root causes of disengagement can only be done by studying users’ rationales for not using a security mechanism, not by studying how they, or others, fail to use it when they already want to.”

With the discussion, above it was highlighted that there is a difference in perceptions concerning the existence of trade-offs between security and usability. The divided opinions of the community pose a challenge imperative to be addressed.

3 Interview Protocol and Findings

3.1 Data Collection

The data was collected using 12 semi-structured interviews and discussions with members of the 2 leading product lines of a major European software development organization. The participants included product owners, architects, developers, security engineers, and UX developers. The interview had 3 major themes: (1) how are security and usability aligned during the development lifecycle of the products, (2) who handles the conflicts between security and usability during the development, and (3) how it is ascertained that the product is secure AND usable? The interviewees also shared instances of the conflicts they encountered in their product lines and challenges faced in the alignment between security and usability. Each interview lasted approximately one hour. The interviews were audio-recorded for analysis purposes and due ethical concerns were considered in this regard. The interview data was later transcribed and co-related with the notes taken by the researcher during the interviews.

3.2 Analysis Methodology

The interview data was analyzed using the Gioia method [14]. The Gioia method is a qualitative data analysis method with an inductive approach. One of the reasons for

its choosing was its inductive nature as it allows making broad generalizations based on informants' understanding of the organizational events. In line with the specifics of the Gioia method, a 3-stage analysis method was followed. In the first stage, the interview transcripts were read thoroughly followed by listening to the audio recordings of the interviews. The intent was to assign first-order codes to the interview data. Codes were assigned to repeated statements, surprise responses, aspects stressed by the interviewees, or something similar as reported in the previous studies, and related to some theory/model. Table 1 shows the codes created during this stage along with the example quotes by the interviewees.

After this exercise, the first-order trends were finalized. In the second stage, the related codes were merged to develop broader categories and abstract concepts. Finally, in the third stage, the second-order concepts were aggregated to form broader themes relevant to alignment between security and usability in the industry. The second-order concepts and the aggregated themes are presented in Fig. 1.

3.3 Findings

With reference to the content presented in Sect. 3.2, after analysis of the interview data, two aggregated concepts were identified as (1) current gaps in the management of the conflicts, and (2) consideration for future solutions. Current gaps in the management of the conflicts relate to gaps in the industrial practices and procedures regarding alignment between security and usability. These gaps include:

- less emphasis on usability as compared to security, despite the fact that both usability and security are equally desired characteristics in software systems.
- there are conflicts between security and usability the trade-offs always favoring security.
- there are no designated roles for management of the conflicts, the roles vary across different teams.
- there are no formal communication mechanisms between the security and usability teams for concerns to be integrated from both sides,
- usability aspects are not well integrated into the design and development phase of the systems and services.
- there are no existing practices and methods that guide the developers in the management of conflicts.
- there is no specified phase in the product development lifecycle for management of the conflicts, although the interviewees agree the earlier the better management of conflict approach, in practice it's often late in the product development lifecycle.

Furthermore, it was identified that the use of design patterns can help the developers in the management of conflicts more effectively. The idea is to support the developers in handling security and usability conflicts by using the design patterns. Patterns provide benefits like means of common vocabulary, shared documentation, and improved communication. Also, the pattern can be incorporated during the early stages of system development in contrast to considering usability and security later in the development

Table 1. Key concepts and associated codes during stage 1

Concepts	Example of codes	Example quotes
Security and usability are inter-related	Conflicts, value, reputation, trade-offs	<p>“Security is really important, so is usability, bad UX can lead to bad security.”</p> <p>“Security is very important from a management perspective; however, usability can help bring competitive advantage. Yes, there are conflicts between the two.”</p>
Security is more important than usability	Weight, competitive advantage, cost, value	<p>Security is most important for the whole product both in terms of value and reputation of the company, usability is very important, but security has more weight.”</p> <p>“Security is critical to ensure that the data remains safe, “Do the secure things, if not easy then the next easiest thing”.</p> <p>Usability is required to serve need/business goals. There are conflicts.”</p>
Lack of formal communication mechanisms between teams	Discussion, informal communication, issue-specific results	<p>“There are different roles for handling usability and security in a project and there are no communication mechanisms specifically for usability and security developers. It’s the same as all others.”</p> <p>“There are different teams for both and discussion is done when there are issues.”</p>
Security has the final say	Usability aspects, integration of concerns	<p>“There is a discussion for communication, it is informal, and security has the final say.”</p> <p>“There are different roles for each aspect, UX people sketches are discussed, security people raise a hand to change. Does not happen the other way round.”</p>

(continued)

Table 1. (continued)

Concepts	Example of codes	Example quotes
Frequency of occurrence of the problem	Every day, often, repetitive, commonly	<p>“Frequently encountered conflicts, they are repetitive, when trying to solve security issues it adds usability issues in the product.”</p> <p>“Commonly encounter security and usability conflicts, especially when starting to design new systems.”</p>
Lack of practices of methods for handling conflicts	Practices and methods, informal communication	<p>“No practices and methods used for handling usable security exist, discussion-based approach is used.”</p> <p>“No practices and methods used for handling usable security exist, informal communication.”</p>
Different roles involved in the process	Developers, product owners, UX specialists, security engineers, architect	<p>“Developer and product owner handle the conflicts in case they arise.”</p> <p>“UX specialist, security engineer, product lead architect discuss, and final verdict often favors security over usability.”</p> <p>“Product owner, architect discuss. Trade-offs are situational security is very critical.”</p>
No specified phase in the development life cycle during which conflicts should be handled	Requirements, design, implementations, testing	<p>“Should be during the requirements and design but does not happen often, its worst when it happens during the QA and testing.”</p> <p>“Ideally should be during the design phase, but currently during the implementation and testing phase”</p>

(continued)

Table 1. (continued)

Concepts	Example of codes	Example quotes
There is a business impact of compromise on usability due to security	Number of users using the system, business impact	“Usability of security could impact the number of users using the system.” “It does have a business impact.”
No efforts and costs were spent on engineering the conflicts	Not determined, very little, not measured	“No cost and effort spent on engineering the conflicts, if it is there it’s very small but there should be.” “Very tiny/not applicable sometimes, there is already a security framework no deviations allowed.”
Not assessing the usability of security features	No means, metrics, independent assessments	“There is no means to assess the usability of security.” “Independent assessments are done, nothing for usable security.”

lifecycle. It is perceived that handling the usable security problem earlier in the development lifecycle will help in saving significant costs and delays associated with re-work. Moreover, patterns’ ability to be improved over time and incorporate multiple viewpoints make them suitable for interdisciplinary fields like usable security [5].

Patterns can be effective in assisting the developers in making reasonably accurate choices while dealing with conflicts. Each pattern expresses a relation between three things, context, problem, and solution. Patterns provide real solutions, not abstract principles by explicitly mentioning the context and problem and summarizing the rationale for their effectiveness. Since the patterns provide a generic “core” solution, their use can vary from one implementation to another. A usable security pattern encapsulates information such as name, classification, prologue, problem statement, the context of use, solution, and discussion pertaining to the right use of the pattern. More details on how a usable security pattern looks like are presented in [5]. A challenge in this regard is collecting such patterns and making a catalog to be disseminated to the developers.

In addition, it was also identified that there is a need for metrics for the assessment of the usability of security. Usability-only measurement strategies do not hold equally good for the usability of security systems [9]. There is a need for the development of metrics for measuring the adequacy of usable security. To do so, there can be two options: (1) develop a set of usable security metrics, and (2) evolution of the existing usability evaluation metrics to hold good for measurement of security. In this regard, the evolution of the existing usability metrics seems to be a more feasible option. For instance, one such metric could measure the degree of conflict between sub-characteristics of security and usability, respectively. Moreover, in usable security research, there has been an

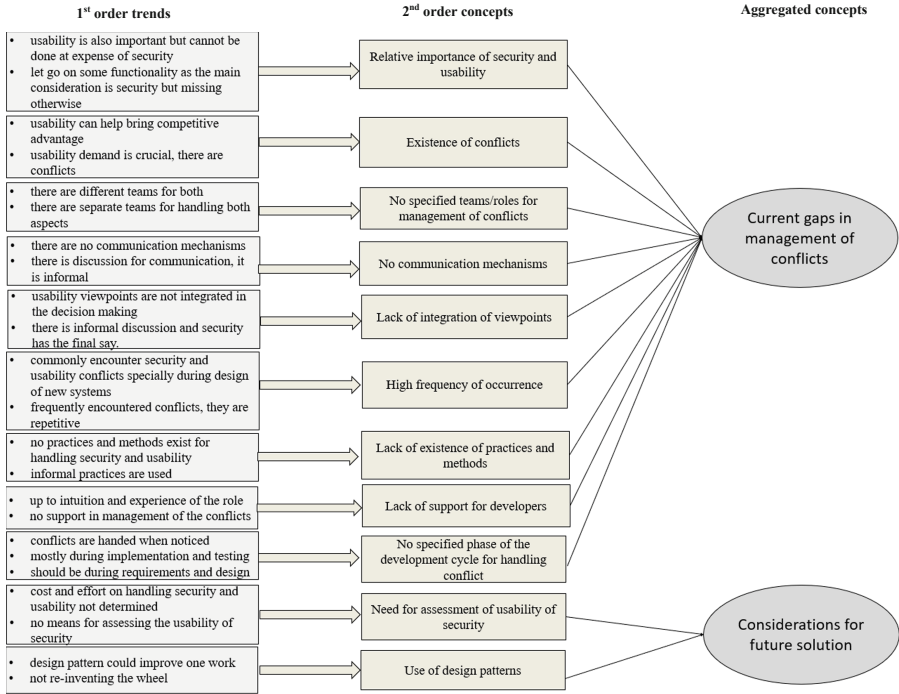


Fig. 1. Challenges in alignment between security and usability in the industry

emphasis on determining the deviation from the user's primary task, which would also require a set of metrics to determine such a deviation. A measurement methodology [15] identifies metrics such as NUC (number of user complaints). However, the efficacy and completeness of the set of such metrics is something that needs to be explored further.

4 Framework for Addressing the Challenges

Based on the challenges identified after the interviews, the framework presented in Fig. 2 was created based on the elements of design science research (DSR). Design science research is a method focused on the development of artifacts to solve existing problems. DSR has a dual mandate: (1) it attempts to generate new knowledge, insights, and theoretical explanations, and (2) it allows the utilization of existing knowledge to solve problems and improve existing solutions [17]. Design science attempts to create artifacts that serve human purposes [16].

The framework has been developed considering its application in agile development contexts specifically Scrum. Though, the framework is inspired by the work [4]; the difference lies in the fact that some of the stages have been left out to support the agile development model. It is relevant to mention that the framework after its creation was subjected to validation from the interviewees during a post-interview workshop. The workshop was held online where the challenges identified after the interviews were

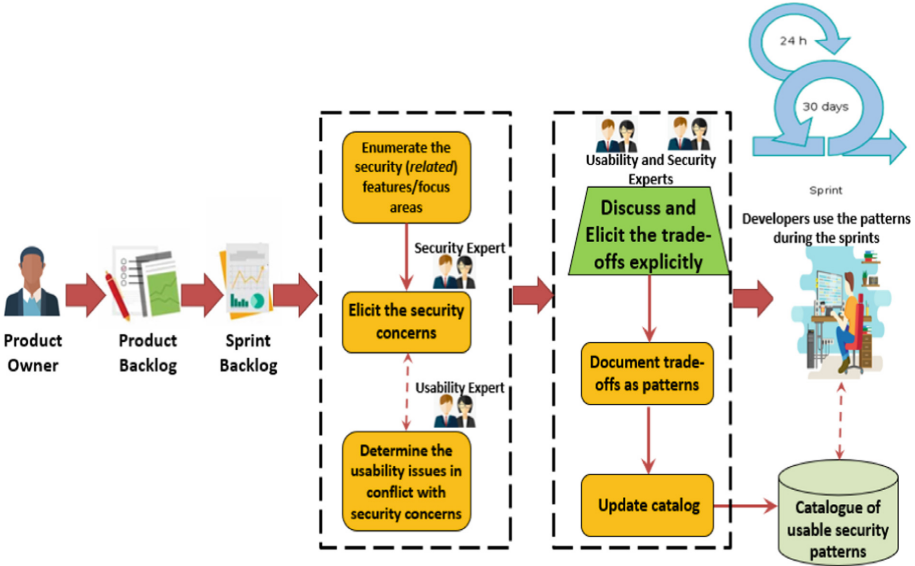


Fig. 2. Framework for aligning security and usability in agile development contexts

presented and the framework was proposed as a potential solution. The participants agreed that the framework has the potential to contribute to addressing the challenges faced by the industry.

The framework addresses the challenges of no designated roles for management of the conflicts by assigning different responsibilities to different roles, for instance, the product owner (scrum master) maintains the product backlog based on which the sprints are planned. Other roles and activities as presented in Fig. 2 are discussed as follows.

- *Enumerate the security features and focus areas*: The security experts working on the project assess the sprint backlog to identify the security requirements. This is done to ensure a specific focus on requirements directly affecting security and its usability.
- *Elicit the security concerns*: For the enumerated security requirements, a specification of what is required from the security point of view is explicitly identified by the security experts. This involves the identification of affected sub-characteristics of security (including confidentiality, integrity, and availability, among others). While eliciting the concerns, it is important to consider both internal and external threats.
- *Determine the usability issues in conflict with security concerns*: Once the security concerns are known, the requirements associated with each of the security concerns are subjected to usability analysis to identify instances of potential conflicts. A matrix of sub-characteristics of security (rows) and sub-characteristics of usability (columns) are created (see Fig. 3). Each element of the matrix describes a potential conflict.
- *Discuss and elicit the trade-offs explicitly*: Once the security and usability concerns are known, the trade-offs are elicited explicitly with the objective of having minimum possible compromise to any of the characteristics and their relevant sub-characteristics. For eliciting the trade-offs, the security and usability experts can use (1) goals from

the security and usability perspectives identified earlier, and (2) standards and best practices concerning security and usability. This may sound like an optimistic approach but by integrating concerns from both perspectives minimal trade-offs have been achieved, practically the example for these includes a single sign-on where a client after the first sign-in can access different systems without having to sign in each of them.

- *Document trade-offs as patterns*: The trade-offs thus identified are documented as design patterns. The patterns can then help other developers solve security and usability alignment issues occurring in similar contexts. More details on patterns’ documentation as well as the example of usable security patterns are presented in [4, 5].
- *Update catalog*: Whenever a new design pattern is documented, it is added to the catalog. This has two advantages, (1) developers working on different projects can use these patterns in case they face the same problem with a similar context, and (2) the patterns enter their validation and evolution phase where it is subjected to validation and comments by other developers who use it, and in case it does not serve the needs, the solution proposed by the pattern can be updated or a new pattern can be documented.

Furthermore, the approach targets to address the gaps identified after interviews. The gaps addressed include no specified teams/roles for management of conflicts, no communication mechanisms, lack of integration of viewpoints, lack of existence of practices and methods, and lack of support for developers, among others. Moreover, it also captures the considerations for future solutions by documenting the identified trade-offs as patterns. However, for the assessment of the usability of security solutions, the framework partially considers this aspect due to fact that patterns evolve with time and as better solutions are identified the patterns can be updated, however, the need for metrics (for instance) for measurement of the degree of trade-offs is something which needs to be considered as part of the future work.

	Security	Usability		
		Effectiveness	Efficiency	Satisfaction
Confidentiality				
Integrity				
Availability				
Authentication				

Place an "X" in the cell where there is a potential conflict

Fig. 3. Matrix for describing a potential conflict at a sub-characteristic level

5 Conclusion

This paper presents an analysis of the state of the art considering the alignment between security and usability in the industry. The paper presents findings after conducting a series

of semi-structured interviews with different roles at a leading European development organization. The interviews identified several gaps in the state of the art including no specified teams/roles for management of conflicts, no communication mechanisms, lack of integration of viewpoints, lack of existence of practices and methods, and lack of support for developers, among others. The paper also presents a framework to be incorporated during the product development lifecycle for improving the current state of the art. It is worthwhile to mention that the version of the framework presented in the paper was validated during a post-interview workshop conducted with the interviewees.

References

1. Sasse, A., Rashid, A.: The Cyber Security Body of Knowledge — Human factors knowledge area v 1.0. The University of Bristol (2019). https://www.cybok.org/media/downloads/Human_Factors_issue_1.0.pdf. Accessed 23 Dec 2021
2. Garfinkel, S., Lipford, H.R.: Usable Security: History, Themes, and Challenges. Morgan & Claypool Publishers, USA (2014)
3. International Standardization Organization (ISO) (2011). Systems and software engineering – systems and software quality requirements and evaluation (SQuARE) – system and software quality models, ISO 25010
4. Naqvi, B., Clarke, N., Porras, J.: Incorporating the human facet of security in developing systems and services. *Inf. Comput. Secur.* **29**(1), 49–72 (2020)
5. Naqvi, B., Seffah, A.: Interdependencies, conflicts, and tradeoffs between security and usability: why and how should we engineer them? In: 2019 1st International Conference HCI-CPT held as part of the 21st HCI International Conference, HCII 2019, pp. 314–324 (2019)
6. Garg, H., Choudhury, T., Kumar, P., Sabitha, S.: Comparison between significance of usability and security in HCI. In: 2017 3rd International Conference on Computational Intelligence Communication Technology (CICIT), pp. 1–4 (2017)
7. Barlev, S., Basil, Z., Kohanim, S., Peleg, R., Regev, S., Shulman-Peleg, A.: Secure yet usable: protecting servers and Linux containers. *IBM J. Res. Dev.* **60**(4), 12:1–12 (2016)
8. Bai, W., Kim, D., Namara, M., Qian, Y., Kelley, P.G., Mazurek, M.L.: Balancing security and usability in encrypted email. *IEEE Internet Comput.* **21**(3), 30–38 (2017)
9. Wang, Y., Rawal, B., Duan, Q., Zhang, P.: Usability and security go together: a case study on database. In: 2017 Second International Conference on Recent Trends and Challenges in Computational Models (ICRTCCM), pp. 49–54 (2017)
10. Nwokedi, U.O., Onyimbo, B.A., Rad, B.B.: Usability and security in user interface design: a systematic literature review. *Int. J. Inf. Technol. Comput. Sci.* **8**(5), 72–80 (2016)
11. Kulyk, O., Neumann, S., Budurushi, J., Volkamer, M.: Nothing comes for free: how much usability can you sacrifice for security? *IEEE Secur. Priv.* **15**(3), 24–29 (2017)
12. Sasse, M.A., Smith, M., Herley, C., Lipford, H., Vaniea, K.: Debunking Security–Usability Tradeoff Myths, p. 7 (2016)
13. Dodier-Lazaro, S., Sasse, M.A., Abu-Salma, R., Becker, I.: From paternalistic to user-centered security: putting users first with value-sensitive design. In: CHI 2017 Workshop on Values in Computing, p. 7 (2017)
14. Gioia, D.A., Corley, K.G., Hamilton, A.L.: Seeking Qualitative rigor in inductive research: notes on the gioia methodology. *Organ. Res. Methods* **16**, 15–31 (2013). <https://doi.org/10.1177/1094428112452151>
15. Naqvi, B., Seffah, A., Braz, C.: Adding measures to task models for usability inspection of the cloud access control services. In: Bogdan, C., Kuusinen, K., Lárusdóttir, M., Palanque,

- P., Winckler, M. (eds.) Human-Centered Software Engineering. HCSE 2018. Lecture Notes in Computer Science, vol. 11262, pp. 133–145. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-05909-5_8
16. Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: A design science research methodology for information systems research. *J. Manag. Inf. Syst.* **24**(3), 45–78 (2007)
 17. Baskerville, R.L., Kaul, M., Storey, V.C.: Genres of inquiry in design-science research: justification and evaluation of knowledge production. *MIS Q.* **39**(3), 541–564 (2015)
 18. Guan, L.: Cyberattack hits regional Victoria hospitals, Patient records and booking system shut down. <https://ia.acs.org.au/article/2019/cyberattack-hits-regional-victoria-hospitals-.html>. Accessed 28 Jun 2022
 19. Avizienis, A., Laprie, J.C., Randell, B., Landwehr, C.: Basic concepts and taxonomy of dependable and secure computing. *IEEE Trans. Dependable Secure Comput.* **1**(1), 11–33 (2004)