R209 Essay: Usable security

Chongyang Shi (*cs940*)

October 18, 2017

1 Summaries of research

In order to determine whether security softwares with conventionally well-designed user interfaces can adequately assist novice users in performing complex security tasks, Whitten and Tygar [1] performed a case study on PGP 5.0. They evaluated the security usability of its UI through two methods: a cognitive walkthrough of confusions facing a novice user of the UI, and user tests conducted with potential users unfamiliar with the software and the PGP concept. Evaluations results showed that PGP 5.0's UI fails to adequately demonstrate the concepts of email encryption and authentication, and various problems in visual design can lead to confused users performing dangerously insecure operations. However, those critical of this study may consider the choice of PGP 5.0 to be sub-optimal for evaluations, as the protocol design of PGP is considerably more complex than other security tools a novice user may come across, such as symmetric file encryption with a pre-shared key [2]. This may have reduced user performance in user tests.

Herley [3] reviewed the state of usable security research under the contemporary internet context, and found that past efforts in making security components of softwares more usable may have been misdirected. The author argued that many past efforts have overestimated the capacity a user has for completing security-related tasks, and could not produce a usable solution accomplishable with the limited time and attention the user could provide. Exaggeration of average-case security risk [3, 2.3] and creating unrealistic burden of security mandates [3, 1.3] can also lead to the user cutting corners on security. Therefore, the author believes that future research on usable security should be efficiency-focused. While this paper effectively summarised the perceived shortcomings in usable security research, it is relatively thin on detailed recommendations to overcome these problems.

Most recently, Glass et al. [4] studied how the orderings and the transitions between components of a security-related task could impact the usability of the task. Based on the real world scenario of using an airline check-in kiosk, transitions between a set of partially-ordered tasks were modelled as a constraint satisfaction problem (CSP), whose optimal solution in ordering tasks correlated with the best user performance in user tests and the favoured solution by experts surveyed [4, Fig. 7]. This verified the viability of cognitive framework established by the authors. Some limitations of the models were recognised by the authors, the most critical of which is the potential lack of consistency with user expectations on how certain tasks are ordered.

2 Key themes of research

2.1 Sell security to users on incentives, not on endless mandates

A theme observed across the three papers is the need to provide users with good reasons to use security, rather than mandating the users with endless security policies that may result in workarounds and corner-cuttings being invented. Limitations and ineffectiveness of security automations and user trainings were observed by Whitten and Tygar [1, Sec. 1]. Herley noted that security mandates may not convince the users that cost of security compliance is outweighed by its benefits, and that some mandates such as periodic password changes accomplish very little [3, 1.3, 2.2]. Glass et al. also found that when the security mandate (in this case, passport authentication) is placed in the least obstructive way within procedures, users are most satisfied with the process [4, Sec. VII].

2.2 Provide the user with minimal information required for security

It was observed by all three papers that when presented with an overwhelming amount of security information, the user tends to perform less well in accomplish the task. During PGP 5.0 user tests by Whitten and Tygar, too much information about PGP keys, automatically generated trustworthiness levels, or even excessive vocabulary could confuse a novice user [1, 4.6, 4.7]. Herley [3, 2.1] believes that security advices have provided users with too much information to follow effectively. The framework of Glass et al. [4, III. D.] also focuses on improving performance by minimising overall task demand.

2.3 Simplify organisational goals to ease mandates on users

A final theme covered to various degrees by the three papers is the need for policymakers and software engineers to simplify their security goals to reduce the amount of mandates on users. While under ideal circumstances users will sign PGP keys of those they trust, Whitten and Tygar [1, 4.4] believe that this is beyond the abilities of novice users, and hence trust levels should not be prominently displayed by PGP 5.0 by default to avoid confusing these users. Herley [3, 2.4] illustrated the example of avoiding requiring users to choose passwords resilient against offline attacks by adequately protecting hashed passwords stored. By simplifying the organisational security goals without compromising a reasonable level of security, user experience with security procedures can be vastly improved.

3 Ideas of current context

Evolving desktop operating systems have provided more flexibility in designing user interfaces and displaying metaphor icons [1, 4.1] since Whitten and Tygar's [1] evaluation of PGP in 1999. The theoretical basis of public key cryptography has also not changed much other than acceptable key sizes. However, modern PGP implementations still fail to achieve good usable security, as assessed by Sheng et al. [5] and Ruoti et al. [6] in 2006 and 2015 respectively. Similar user testings were performed across all three studies of PGP, while improvements

to the interface design were recommended and improved between studies, the fundamental problem of the PGP protocol being hard to understand by novice users persist [6, p. 4], which needs to be addressed by improved user training.

In the relatively recent research, Herley's [3] idea of security problems often caused by users overloaded with security-related tasks rather than by insufficient security measures are now well-concurred. Efforts have been made to reduce the amount of security-related tasks users have to perform everyday. This is represented in organisational settings by single sign-on (SSO) [7, 8.1], and in mobile devices by the introduction of fingerprint unlocking [8] and even more recently face-recognition unlocking [9].

An alternative interpretation of the persistent difficulties faced by novice users of PGP, supported by Herley's [3] argument that difficult security mandates may overwhelm users is that certain security measures may be fundamentally beyond the target user group, due to them being inherently too complex. This has been studied with a complexity framework by Benenson et al. [10]. This could mean that the study of usable security may hit a roadblock without developing new security protocols that are more easily understood by novice users.

4 Literature review

In addition to the two further usability reviews of PGP implementations since 1999 [5, 6], there have been research efforts in replacing the knowledge-based authentication systems (along with its flaws in usability and user knowledge) all together, replacing with recall-based authentication with promising results [11]. However, significant shortcomings in recall-based authentication have been noted [12], making practical use difficult.

As with the aforementioned complexity study [10], Gerck [13] also found that the technical complexity of PGP constrains security usability, and alternative protocol designs may be required. Further to Herley [3], discrepancies in security practices of expert and novice users have also been noted by Ion et al. [14]. Some practical means of educating users about security devices such as PGP have been recommended by Redmiles et al. [15].

A key component of the cognitive framework by Glass et al. [4, III. A.] is the prediction of time required to complete a given task, which can be based on CogTool [16]. CogTool is widely used in cognitive modelling, such as evaluating helicopter interfaces [17] and keystroke on handheld devices [18]. The use of constraint satisfaction (CSP) in cognitive modelling has also been previously conducted to model air traffic controller behaviours [19] and human in virtual environments [20].

(1205 words according to texcount.)

References

[1] A. Whitten and J. D. Tygar, "Why johnny can't encrypt: A usability evaluation of pgp 5.0." in *USENIX Security Symposium*, vol. 348, 1999.

[2] S. Gujrati and E. Y. Vasserman, "The usability of truecrypt, or how i learned to stop whining and fix an interface," in *Proceedings of the third ACM conference on Data and application security and privacy*. ACM, 2013, pp. 83–94.

[3] C. Herley, "More is not the answer," *IEEE Security & Privacy*, vol. 12, no. 1, pp. 14–19, 2014.

[4] B. Glass *et al.*, "The usability canary in the security coal mine: A cognitive framework for evaluation and design of usable authentication solutions," *arXiv preprint arXiv:1607.03417*, 2016.

[5] S. Sheng *et al.*, "Why johnny still can't encrypt: evaluating the usability of email encryption software," in *Symposium On Usable Privacy and Security*, 2006, pp. 3–4.

[6] S. Ruoti *et al.*, "Why johnny still, still can't encrypt: Evaluating the usability of a modern pgp client," *arXiv preprint arXiv:1510.08555*, 2015.

[7] M. A. Sasse *et al.*, "The great authentication fatigue–and how to overcome it," in *International Conference on Cross-Cultural Design*. Springer, 2014, pp. 228–239.

[8] I. Cherapau *et al.*, "On the impact of touch id on iphone passcodes," in *SOUPS*, 2015, pp. 257–276.

[9] A. Inc, "Face id security guide," September 2017. [Online]. Available: https://images.apple.com/business/resources/docs/FaceID_Security_Guide.pdf

[10] Z. Benenson *et al.*, "Maybe poor johnny really cannot encrypt: The case for a complexity theory for usable security," in *Proceedings of the 2015 New Security Paradigms Workshop*. ACM, 2015, pp. 85–99.

[11] R. Dhamija and A. Perrig, "Deja vu-a user study: Using images for authentication." in *USENIX Security Symposium*, vol. 9, 2000, pp. 4–4.

[12] S. Wiedenbeck *et al.*, "Passpoints: Design and longitudinal evaluation of a graphical password system," *International journal of human-computer studies*, vol. 63, no. 1, pp. 102–127, 2005.

[13] E. Gerck, "Secure email technologies x. 509/pki, pgp, ibe and zmail," *Corporate Email Management*, pp. 171–196, 2007.

[14] I. Ion *et al.*, ""... no one can hack my mind": Comparing expert and non-expert security practices." in *SOUPS*, 2015, pp. 327–346.

[15] E. M. Redmiles *et al.*, "I think they're trying to tell me something: Advice sources and selection for digital security," in *Security and Privacy (SP), 2016 IEEE Symposium on*. IEEE, 2016, pp. 272–288.

[16] R. Bellamy *et al.*, "Deploying cogtool: integrating quantitative usability assessment into real-world software development," in *Proceedings of the 33rd International Conference on Software Engineering*. ACM, 2011, pp. 691–700.

[17] J. Ludwig, "Comparing helicopter interfaces with cogtool," in *7th International Conference on Cognitive Modeling, Trieste, Italy*, 2006.

[18] L. Luo and B. E. John, "Predicting task execution time on handheld devices using the keystroke-level model," in *CHI'05 extended abstracts on Human factors in computing systems*. ACM, 2005, pp. 1605–1608.

[19] A. Cerone *et al.*, "Formal analysis of human-computer interaction using model-checking," in *Software Engineering and Formal Methods, 2005. SEFM 2005. Third IEEE International Conference on*. IEEE, 2005, pp. 352–361.

[20] S. Smith and D. Duke, "Using csp to specify interaction in virtual environments," *REPORT-UNIVERSITY OF YORK DEPARTMENT OF COMPUTER SCIENCE YCS*, 1999.