Improving the Usability of Security Features within Tools and Applications

C.Heeren and S.M.Furnell

Centre for Security, Communications and Network Research,
Plymouth University, Plymouth, UK
e-mail: info@cscan.org

Abstract

A predominantly observation based study utilising eye gaze tracking and a 'think-aloud' protocol to discover general computer user's perceptions and behaviour towards security. Participants interacted with two security tools, each within a specific task-based scenario while their screen session and eye gaze were logged. Qualitative findings analysed to document each participant's perceptions, leading to usability recommendations for future security tools.

Keywords

Security, Perceptions, Usability, Think-aloud, Gaze Tracking

1 Introduction

With growing risks of increasingly sophisticated attempts to attack vast numbers of systems on-mass, and the evolution of attacks becoming more passive and discreet from the user (Richardson, 2008), typical users who may not have such a comprehensive understanding of computer security specifics need for any security related software to be easy to derive understanding from, manipulate, and function equal to the intension inferred by their interfaces.

The intermediary of the security interface is where perceptions are made of what is occurring in front of them, what they feel they should be made aware of, and how the system is reacting to their interaction. User satisfaction regarding a system may be directly affected by the amount in which a user is able to gain understanding from their interaction. It has been recommended that to align usability and security they must be included as goals throughout the iterative design process, and not an afterthought (Yee, 2004). Lacking the matching of mental models between user perception and what the system actually offers could lead to incorrect assumptions being made as to security processes, therefore producing insecure scenarios (Smith, 2003; Smith, 2008). A security interface may be mentally placing end users outside the active system boundary, facilitating that any negative action is due to user bad practice - ignoring insufficient security design, whereas use-centric design should prevent security compromise solely user (Zurko, 2005).

2 A practical study of usability

A study was conducted with the aim of discovering usability issues which could occur for typical 'everyday' non-technical computer users working with security applications within their home environment. Rather than providing a basis to only find the security knowledge an individual has, and their outright performance upon a security related task such as within previous works E.g. (Katsabas et al., 2005; Helala et al., 2008), provision was made to measure more about a user's outlook and perceptions of what may be occurring with regard to security would be an interesting direction to progress. Issues discovered and perceptions gathered were to be developed into guidelines for methods of security usability improvement. 8 adult participants were evaluated, with their only criteria being that they were regular computer users but not specific technical users. The study took the following design:

Part 1: Completion of a background questionnaire

Part 2: Monitored Security Task Scenarios (Two scenarios, each with six specified tasks, which may be faced within two general and publically available security tools:

Scenario 1: Took place using BitDefender Internet Security 2010 (v13.0.21 during testing), assessed as an affordable yet comprehensive security system which compromises of a deep packet firewall, anti-virus, anti-spyware, and anti-phishing tools (BitDefender, 2010) It appeared understandable for less aware users, having the ability to provide varying levels of interface complexity and control. Participant were faced with having to configure the security system as if it was just installed, with the measures they thought appropriate and to complete the tasks provided as best as possible. Tasks included managing vulnerabilities, firewall manipulation, and selecting an appropriate user interface profile.

Scenario 2: This involved using an encryption package called Advanced Encryption Package Professional v5.3.8 (AEPPro, 2010), which would require the participant to make use of text encryption for the scenario of sending and receiving email needing high privacy. They were to establish a public and private key set, encrypt their message, and email the message to a fictional contact (created by the researcher). Participants then were required to correctly extract a received encrypted message, along with key, and decrypt it so that they may read it. This test was devised due to the previous encryption difficulties found by (Whitten & Tygar, 1999). Although the process appears to be improving in comparison to these earlier findings, AEP Pro was chosen because it was felt it still demonstrated complexity in usability which would provide a good context to observe more participant perceptions than a tool nearer full automation.

Part 3: A usability feedback form to assess their impressions of both of the security tools used, and one perception question for each scenario which was faced, regarding an important part within the scenario.

Part 4: A set of questions regarding their interpretations of the software used.

Throughout: Observations of interesting aspects of each participant's progress, key mistakes or points of confusion, and any discussions or comments they make about the security tasks, and how the software makes them feel.

Finally: The researcher analyses the recorded materials of the participant's test session and scores each session according to the usability metrics listed, prior to further analysis regarding the observations made, and the behavioural data recorded.

HyperCam 2 v2.23.01 (Hyperionics, 2010) was used for screen recording the participant's sessions. For the process of gaze tracking a bespoke headset was created and gaze tracking was detected and logged as the participant's session via Gaze Tracker v1.5.0.211 (Gaze Tracker, 2010). Participant gaze points were utilised in Ogama v3.3 (Ogama, 2010), a usability analysis tool capable of producing visualisations of gaze data upon user sessions.

3 Findings and Recommendations

The key findings arising from the study are summarised in the following paragraphs:

1. Help topics which are as specific as the request: A satisfying sight for a confused user could be an exact help topic being found for the issue you searched for, first time. A participant, for example, would check through help, not find an exact matching topic even though other results were returned, and exit to resume the same struggle they were confronted with earlier. Perhaps a level of contempt for poor previous help facilities has made many users reluctant try harder once more? One participant commented on online help and program help as being of "not much help".

2. Helpful information should be in the right places: Looking to generate encryption keys through a help search finds no mention of the menu location under 'Generating key files', whereas it is clearly documented under 'Introduction' (Figure 1), assuming you attempted long enough to find it before giving up. The information may be relevant in both, but at least include it under its own topic, as well. You may think it wise to add each of these discreet help topics, but not if they fail to contain some of the most important information required. One participant was found to take almost half of their encryption task completion time referring to help.

Inadequate and insignificant navigation within BitDefender was also found. Figure 2 displays the gaze tracking for a participant attempting to locate the 'Add Rule' button within Firewall controls. The buttons are insignificant and within the top-right area of the window, where there is a distinct lack of gazing (Figure 2).

3. Inappropriate or inconsistent terminology: Would a typical security user, or even an advanced user for that matter, expect a key '*Comment*' to be mandatory? Within the situation of key generation within AEP Pro it results in actually being the prefix for both file names (as it more subtly states in brackets). Many participants hardly noticed this input box until prompted that the value was missing. To further compound confusion, no explanation of why the comment is needed was ever offered (Not before, in the form of a tooltip, during, as part of an admonition (Yee, 2004), or after as information upon the error message).

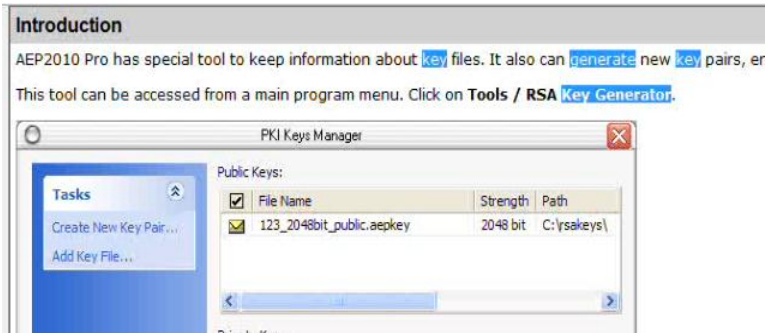

Figure 1: An instance where unhelpful help occurs, prolonging the participant's search.

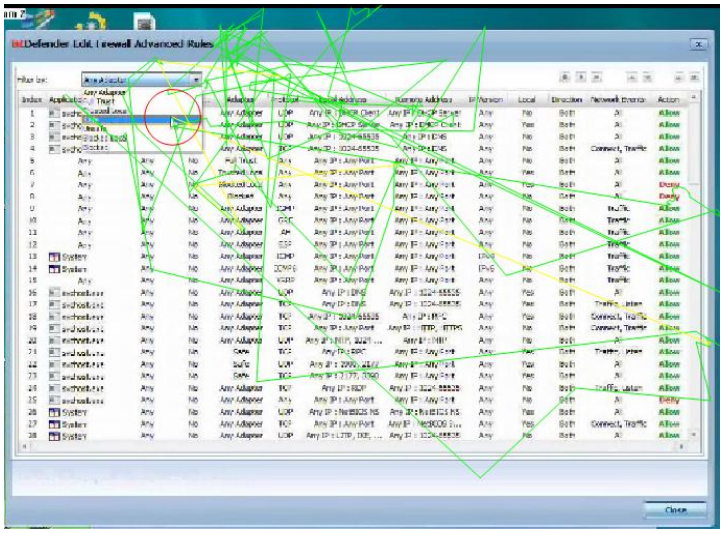

Figure 2: An example of a participant's gaze estimation during a Scenario 1.

One participant even believed that this input box was for the text they wanted to encrypt, and for the combination of password and comment to make keys – An example of the misdirection a system can bestow upon a less technical user.

Another participant thought of the actual task they needed to achieve. They searched help for the relevant topic. This lead them to a screenshot of the menu required, complete with the menu title (Figure 3, left hand) - we are no longer looking for 'generate', but 'create'). This participant appeared to always attempt to match what they saw in help to what they could find upon the interface, as expected you may think. However, within the menus we now have returned numerous re-examinations of more than one help topic simply because this inconsistency is not in keeping with the participant's line of thought. This mismatch appears bring about many of the usability issues mentioned.

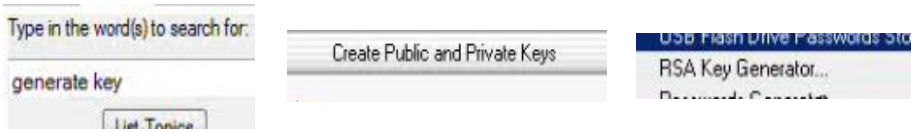

Figure 3: Changing terminology through tasks of Scenario 2

4. Avoiding compounding options with terminology: Even if you later are to include more advanced computing or security specific terminology to your windows, keeping this to a minimum or removing it all together from menu or navigation selections can aid. On the opposite outlook, some expert users may want to seek out specifically termed options, but placing them within a submenu or upon a configurable window itself could reduce the aversion less knowledgeable users suffer towards terms they do not understand.

Using technical prefixes and suffixes upon your menu options appear to result in reluctance to try them by unfamiliar and inexperienced users. Participants would repeatedly browse the same menu, choosing options above and below, even stopping to ponder whether terminology-laden options were what they were seeking, and still declined from trying them. 38% of participants displayed signs of being reluctant to click upon 'RSA Key Generator' simply because they do not understand what RSA means in this context. More than one participant voiced this feeling during testing. Observations appeared to show that rather than experimenting with where that option would lead, some participants would remain hovering over action, with no guidance provided, as if trying to second guess what the outcome of using that selection could mean – before avoiding it.

5. Failing to match the user's mind-set: Generating one pair of encryption keys at one time may not be everyone's way of working, but it is suspected that users wanting to quickly encrypt some text for email want just the one set of keys for this sort of event. AEP Pro's generation interface continued to display the three options 'Create, Cancel, Help' even after key pair creation. Participants were left wondering what was expected from them as little information was provided to signify their task was a success. For many computer users now days, the sight of confirmation being about as descriptive and inconclusive as an *'OK'* placed with the output list, is an very rare or unseen sight. With a seeming lack of respect for a non-technical user, it echoes command-line confirmation typically dating back three decades previous.

However, the main factor here was their inability for the security's design to have reflected the thought process of any unfamiliar user at this time. With thoughts such as, having generated their keys, why isn't the menu closing? - As often expected now days. Even if this wasn't appropriate, no information guides the participant that they may exit the menu, and the buttoned options remain unchanged - so taking on a misleading interpretation for the participant. Will my keys fail if I leave? Why does it say 'Create' when I just created them? Did I actually create them? What about the use of 'Cancel'; surely users have been trained to think that cancel ends an incomplete task, as their interpretation of it within the real world would concur? Here it becomes the only successful avenue of exit from this window, whether the task was successful or not.

6. Visual depictions of tasks are still well received: Regardless of user background, visual metaphors for the tasks the user wants to achieve appear to help improve understand-ability (insuring that they are used with a mind-set matching the user is a different issue) for a typical user. For example, encryption key management for one participant: No prior knowledge of the keys concept was known, yet through just imagining the task achieved by using either of the depicted states of the envelopes, which key governed which purpose was deduced. For another participant the icon also showed the purpose of the object in question, rather than what is was - a key.

Figure 4: Icons displaying the objective rather than object

7. Safe-staging availability seems necessary: If a 'safe-staging' style of task guidance is not provided, it appears that users may attempt a similar style of working themselves. It was found that 50% of participants exhibited signs of attempting to produce their own form of 'self-safe-staging' when no highly usable guidance appeared to be provided, as an attempt to manage a complicated task in smaller segments. Between each of these users, resorting to the standard in-program help, there was a noticeable effort to survey each pictorial element of relevant help upon half of the screen while attempting to track down which navigation options led to the same factor. It appeared to the researcher that this was similar to the actions which occur during safe-staging and the guidance regularly seen beside tasks to be accomplished.

4 Conclusions and future work

Background questionnaire data suggested that many users are now becoming more aware that threats to their actions are a reality. However, many inexperienced, or rather non-technical users who have little to do with a computing field outside of completing everyday tasks with the aid of a computer can be rather left behind and perhaps reluctant to delve into the correct use of security features which make little sense to them (Chatziapostolou & Furnell, 2007).

The findings showed participant perceptions such as, the belief that *'the type of security profile selected for a tool's interface reflected the level of security protection offered',* and appears to be a regard many general users studied have expressed, with added opinions along the lines of *'more 'expert' users require higher security'* aired

by a number, whereas others appear to suggest perceptions along the lines of *'lowest users should have maximum security'*.

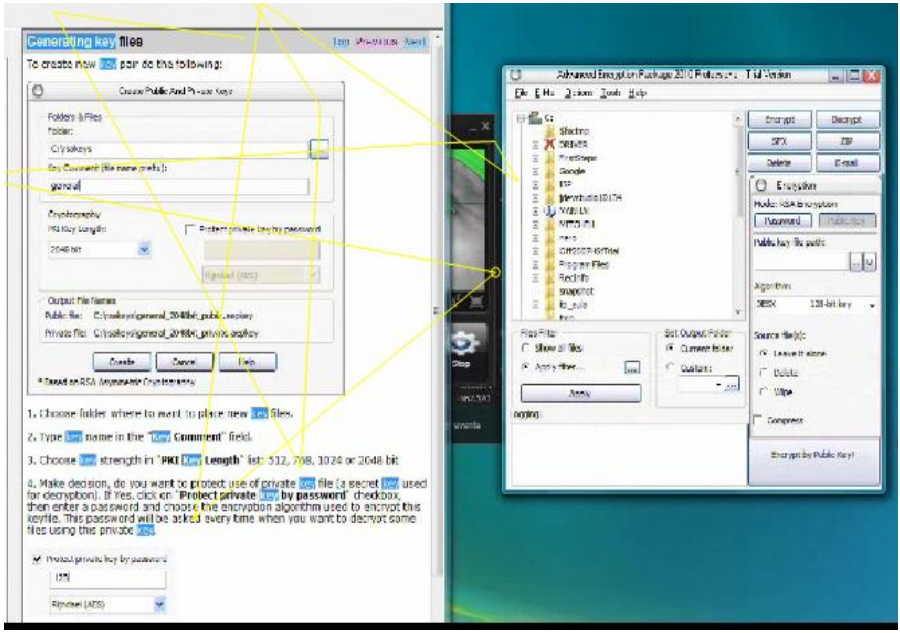

Figure 5: A participant attempting to 'self-safe-stage' his efforts of scenario 2

This might form an expectations gap between what users expect and what is offered - also how users are guided, yet also looked after. Would it mean that an expert carries out difficult or sensitive tasks, and so requires very high security (and can deal with it?) or would it mean that novice users should be looked after the most (perhaps restricted so greatly, they could result in being annoyed towards the security)? Out of the relatively small sample of participants questioned, these perceptions have already both been noted.

The question may remain as to what would be the ideal security solution, this may be complete and 'perfect' automation, yet even if this were possible, may not guarantee protection about threats even analysts, designers, and users are unaware of. As threats become more subtle towards the user, this may be the only way forward for users with less concern or ability to try and understand more about computer security. A criticism could be that this novice restriction combined with a lack of knowledge on the user's part could lead to a false sense of confidence, in that all the software operations are working optimally (as seen in Bit Defender by green ticks and system reassurance even when major security options are disabled, because the tracking of their status is also disabled), and all choices are the best suited to this user's needs – when this may not be the case.

A user lacking knowledge may feel unaware of how secure their computer is at a 'Novice' level, yet unaware of how to secure their computer at 'Expert' level.

As a suggestion, alongside the learning abilities of security software such as firewalls to establish regular required processes, perhaps the inclusion of an artificial training agent which can assess the level of operational flexibility each user requires and could implement scaling of security control to allow for less distinct changes of profile to be undertaken by users. Participant recommendations, such as the use of a 'Tree View' interface deciding this scale of control with interpretation to the security tasks required may provide greater usability for users who feel more reluctant to engage in 'inefficient' use of their time, or who lack a greater level of specific understanding.

5 References

AEPPro (2010): Advanced Encryption Package Professional Software; Available at: http://www.aeppro.com/ [Date accessed: 16th June 2010]

BitDefender (2010); BitDefender Total Security Software; Available at: http://www.bitdefender.co.uk/solutions/total-security.html [Date accessed: 15th June 2010]

Chatziapostolou D., Furnell S. M. (2007) "Recording End-Users Security Events: A Step Towards Increasing Usability" Plymouth University, UK. Available at: http://www.cscan.org/default.asp?page=viewabstract&paperid=385 [Date accessed: 10th January 2010]

GazeTracker (2010); ITU Gaze Tracker open source software v1.5.0.211; University of Copenhagen; Available at: http://www.gazegroup.org/downloads/23-gazetracker [Date accessed: 14th June 2010]

Helala M., Furnell S.M., Papadaki M. (2008) "Evaluating the Usability Impacts of Security Interface Adjustments in Word 2007" Plymouth University, UK. Available at: http://www.cscan.org/default.asp?page=viewabstract&paperid=536 [Date accessed: 14th January 2010]

Hyperionics (2010); HyperCam 2 Software v2.23.01 Available at: http://www.hyperionics.com/hc/ [Date accessed: 9th June 2010]

Katsabas D., Furnell S. M., Phippen A.D. (2005) "IT Security: A Human-Computer Interaction Perspective" Plymouth University, UK. Available at: http://www.cscan.org/default.asp?page=viewabstract&paperid=240 [Date accessed: 14th January 2010]

Ogama (2010); O Ogama open source gaze and mouse analyser v3.3; Freie Universitat Berlin (2010); Available at: http://www.ogama.net/ [Date accessed: 17th June 2010]

Richardson, R. (2008); 'CSI Computer Crime and Security Survey 2008'; Computer Security Institute; Available at: http://www.cse.msstate.edu/~cse6243/readings/CSIsurvey2008.pdf [Date accessed: 7th January 2010]

Smith S. W. (2003) "Humans in the Loop: Human-Computer Interaction and Security" IEEE Security & Privacy; May/June 2003; pp 75-79.

Smith S. W. (2008) "Why do Street-Smart People do Stupid Things Online?" IEEE Security & Privacy; May/June 2008; pp 71-74.

Whitten A., Tygar J.D. (1999) “Why Johnny can’t Encrypt: A usability evaluation of PGP 5.0”; Proceedings of the 8th conference of USENIX Security Symposium; Vol. 8 p14; Available at: http://www.gaudior.net/alma/johnny.pdf [Date accessed: 20th Jan 2010]

Yee, K.P. (2004); 'Aligning Security and Usability'; IEE Security and Privacy; Volume 2; Issue 5 (September 2004); pp 48-55.

Zurko M. A. (2005) “User-Centered Security: Stepping Up to the Grand Challenge”; Proceedings of the 21st Annual Computer Security Applications Conference (ACSAC 2005) 1063-9527/05