Quantifying Users' Beliefs about Software Updates

Arunesh Mathur*, Nathan Malkin†, Marian Harbach‡, Eyal Peer§ and Serge Egelman‡†
*Princeton University †University of California, Berkeley
‡International Computer Science Institute §Consumers Behavioral Insights Lab, Bar-Ilan University, Israel
amathur@cs.princeton.edu nmalkin@cs.berkeley.edu mharbach@icsi.berkeley.edu
eyal.peer@biu.ac.il egelman@cs.berkeley.edu

***Abstract*—Software updates are critical to the performance, compatibility, and security of software systems. However, users do not always install updates, leaving their machines vulnerable to attackers' exploits. While recent studies have highlighted numerous reasons why users ignore updates, little is known about how prevalent each of these beliefs is. Gaining a better understanding of the prevalence of each belief may help software designers better target their efforts in understanding what specific user concerns to address when developing and deploying software updates. In our study, we performed a survey to quantify the prevalence of users' reasons for not updating uncovered by previous studies. We used this data to derive three factors underlying these beliefs: update costs, update necessity, and update risks. Based on our results, we provide recommendations for how software developers can better improve users' software updating experiences, thereby increasing compliance and, with it, security.**

I. Introduction

Software updates are essential to keeping systems and programs up-to-date. These updates fix bugs and bring about improvements in performance and usability; but arguably their most important function is enhancing system security by fixing vulnerabilities. In 2015 alone, Microsoft reported 3,300 vulnerability disclosures of varying threat levels and estimated that close to a quarter of Windows Personal Computers (PCs) were not always protected and updated to the latest patch level [19]. Similarly, Cisco suggested that most security exploits will continue to be propagated by outdated software that contains known vulnerabilities [4]. Therefore, these companies and security agencies—such as the United States Computer Emergency Readiness Team [30]—recommend that users install software updates as soon as they become available, in order to protect systems from being exploited by attackers. In fact, applying updates in a timely manner is one of the few pieces of computer security advice on which experts agree [15].

However, recent studies have shown that users avoid or delay installing software updates [31], [32], [18], [29], [10], [17] and uncovered some of the reasons that users offer for this behavior. While these studies make a timely contribution to aid software developers, they are primarily qualitative. As a result, we know very little about the prevalence of these beliefs. Given that beliefs shape behavior [1], understanding exactly how widespread these beliefs are will help software developers and security professionals more efficiently understand how to address users' concerns and improve the overall software updating experience, and ultimately, affect security.

In our study, we conducted a survey to better understand how updating beliefs vary among the general population, and investigate whether they can be grouped into actionable factors for the community. We contribute to the literature on software updating behaviors by quantifying the prevalence of various beliefs about software updates, as uncovered by the previous qualitative literature. Our analysis uncovers three distinct factors that highlight why users claim to avoid updates:

- *Update Costs*: These costs include the time it takes to install the update, whether a restart is required, and its required space on disk.
- *Update Necessity*: This includes users' satisfaction with the current system or program, whether the update's purpose is clear, and its perceived importance.
- *Update Risks*: These risks include data loss due to the update, and whether the update may be malicious.

We interpret and discuss our results, highlighting broad implications these have for developers who design software updates for users. Understanding these unifying factors may help guide developers towards improving the updating experience through better software design and messaging.

II. Related Work

Failure to patch known security vulnerabilities is one of the leading causes for security breaches; most exploits target systems that have not been patched, rather than undisclosed zero-days [4]. This impacts both end-users and system administrators: for instance, the recent Equifax data breach, the largest of its kind, was as a result of a system administrator's failure to apply a software update [21]. This update fixed a known vulnerability and had been available for several months. Thus, ensuring that people update their systems in a timely fashion would help in improving computer security.

Only recently have security researchers begun exploring how users feel about and respond to software updates. Ion *et al.* [15] compared the advice expert and non-expert computer users gave in order to stay protected online. They found that non-experts installed updates less frequently and lacked awareness about the importance of updates. In a somewhat representative sample of the US, Wash and Rader [33] found that only 24.2% of users reported taking "advanced" security

Network and Distributed Systems Security (NDSS) Symposium 2018
18-21 February 2018, San Diego, CA, USA
ISBN 1-1891562-49-5
http://dx.doi.org/10.14722/ndss.2018.23036
www.ndss-symposium.org

actions—including installing software updates and patches—frequently. Traces of update habits of Android users have shown that only about 50% of all users update to a new application version within the first week of the update's release [22]. Software developers have also been shown to not update third-party libraries after first use [7].

A related set of studies have explored in greater depth why users avoid or delay installing software updates. Rader and Wash [34] found that automatic updates that try to involve users in the decision-making process lead to poor mental models of updating and result in less secure systems. Vaniea *et al.* [32] interviewed 37 Windows users and found that they cited three reasons for avoiding updates: updates introduced undesirable features, the value and purpose of an update was hard to assess, and the need for updating was unclear because the software/program functioned correctly. Mathur and Chetty [17] found that users who have negative experiences with software updating disabled auto-updates on Android devices.

In a survey of 155 users, Fagan *et al.* [9] explored software update notifications and found design features that led to annoying and confusing messages. Mathur *et al.* [18] interviewed 30 users and found they felt annoyed with notifications that interrupted their tasks, including having to restart their machines. They described software updating as an "information problem," highlighting the different pieces and sources of information users sought before updating, including the source of the update and the duration of the install process. Forget *et al.* [10] compared the level of end-user engagement in the management of their computers' security with security outcomes and discovered that greater engagement did not always correspond to greater security. The authors uncovered a variety of reasons why users avoid or delay updates, confirming the results of previous studies [32], [31], [34]. They suggested that security mitigations, such as software updating, need to be designed according to how much users engage with computer security. Vaniea and Rashidi [31] surveyed 307 users about their experiences with updating software and highlighted various user experiences at each step of the process.

Our study builds on this previous work, which uncovered users' underlying beliefs about updating their software, by measuring the prevalence of these various beliefs. Our study uses this to offer practical recommendations for software developers to consider when developing and designing updates in order to deliver a better user experience.

III. Method

A. Survey Construction and Deployment

We first surveyed the literature on why users avoid software updates, noting each belief we came across. Not only does the existing literature provide a breadth of users' beliefs about avoiding updates, but also validates it with actual behavior [10], [32]—that is, users leaving their systems unpatched—in many studies. We discovered a total of 15 different beliefs relating to software updates:

B1-Fine: Updates seem unnecessary because everything works fine [32], [29], [31]
B2-Time: Updates take a long time to install [18], [31]
B3-Restart: Updates require unnecessary restarts of applications or PCs [34], [18], [10], [31]
B4-Unused: Updates are requested by programs used infrequently [32], [31], [18], [10]
B5-Unimportant: Updates seem unimportant [18], [10], [32]
B6-Purpose: The purpose of updates is unclear and hard to understand [18]
B7-Bundled: Updates introduce unwanted bundled programs [31]
B8-Features: Updates add unwanted or remove wanted features from programs [31]
B9-Bugs: Updates introduce new bugs into programs [29], [18]
B10-Space: Updates occupy a lot of disk space [18], [31]
B11-Compatibility: Updates cause compatibility issues between programs [18], [31]
B12-UI: Updates disrupt program user interfaces [32], [31], [18], [10]
B13-DataCost: Updates consume a lot of data on Internet plans [18]
B14-Malicious: Updates contain malicious software [18], [31]
B15-DataLoss: Updates lead to a loss of data [31]

We compiled these beliefs into a survey where we asked participants how often, in their experience, each one of these statements about software updates were true. (Respondents rated their beliefs on a five-point scale, from "rarely" to "always.") We chose not to ask about any specific updating experience (such as specific devices or operating systems), to avoid bias from any recent or otherwise memorable experience. We wanted to collect opinions that people have come to hold about software updates in general, as these would color users' interactions with any new software update. In addition to the ratings, we collected participants' demographic information: age, gender, and (if available) occupation.

We recruited participants from Amazon Mechanical Turk (AMT), an online work marketplace that has been shown to produce diverse and generally representative samples of the population [3]. We limited the survey to only those AMT users who were based in the United States (US) and had a completion rate of 95% or greater. Sampling high-reputation AMT users ensured we did not have to resort to attention check questions [24]. The survey took between 5-10 minutes, and participants were compensated $2 for their time. The beliefs were randomized for each AMT user to avoid any potential ordering bias.

We also launched the survey on Google Consumer Surveys (GCS) to compare the prevalence of these beliefs across two samples. Because the number of questions in our survey (15) exceeded the maximum number of questions GCS allows in a survey (10), we placed each question in a survey by itself. GCS participants were compensated with Play Store credit. This study was approved by our Institutional Review Board.

B. Data Analysis

The beliefs we compiled cover a wide range of the updating process's aspects, and we wanted to uncover any underlying themes that potentially unified multiple beliefs. Rather than

Demographic	AMT Participants	GCS Participants
Age		
18–24	2.49%	14.58%
25–34	40.80%	20.56%
35–44	30.85%	15.54%
45–54	11.94%	12.47%
55–64	8.46%	11.70%
>= 65	5.47%	8.57%
Gender		
Male	58.71%	42.97%
Female	41.29%	42.23%
Other	0.00%	14.8%
Education		
12th grade or less	1.49%	NA
High school	12.44%	NA
Some college	17.91	NA
Bachelor's	45.27%	NA
Associate's	12.94%	NA
Post-graduate	9.45%	NA
Prefer not to answer	0.50%	NA

TABLE I: Demographic Information of the Amazon Mechanical Turk (AMT) and Google Consumer Survey Participants (GCS). GCS does not report data about participants' education levels.

grouping the beliefs thematically, based on our own subjective judgments, we decided to perform principal component analysis (PCA) [14] with Varimax rotation on the responses of participants from our AMT sample. This technique has previously been used extensively in psychology and human-computer interaction literature [28], [16], [25], [12], as it yields more descriptive factors which are robust to correlation. However, we also confirmed our factor loadings using an Oblimin rotation. The resulting dimensionality reduction helps unveil which beliefs are frequently held together, and allows us to explore the factors underlying why users avoid updating their software.

Because we were interested in factors resulting from the prevalence of a belief (and not its strengths), we transformed the original belief scores—measured on a 5-point scale—to a binary variable for our analysis. We encoded "never (1)," "rarely (2)," and "sometimes (3)" as 0—indicating that users did not hold that belief, and encoded "often (4)" and "always (5)" as 1—indicating that users held that belief.

IV. Findings

We received 200 complete responses from AMT and, on average, close to 200 responses for each belief on GCS. Our participants were roughly balanced in gender and age; detailed demographics can be seen in Table I.

A. *Comparing the Two Samples*

We first compared the AMT and GCS responses, conducting a Mann-Whitney U test to examine whether participants' beliefs differed across samples; because we conducted 15 such tests, we applied the Bonferroni correction, and therefore only report significance at $p < 0.003$. The distributions of the beliefs across the two samples are presented in Figure 1.

We found that both samples were mostly in agreement with respect to their beliefs. The differences between AMT and GCS responses were statistically significant for four of the software updating beliefs [B7 ($r = 0.22$), B12 ($r = 0.17$), B14 ($r = 0.20$), B15 ($r = 0.17$)], with small ($r = 0.1$) to medium ($r = 0.3$) effect sizes [5]. Participants in the GCS sample had a stronger belief that:

- updates introduce unwanted bundled programs into software (B7),
- updates disrupt the user interface of programs (B12),
- updates contain malicious software (B14), and
- updates lead to a loss of data (B15).

For all other beliefs, we found no significant differences in prevalence between the two samples.

As described above, the limitations of the GCS platform required us to ask about each belief in a separate survey. These surveys are anonymous, preventing us from examining beliefs across individuals. The GCS sample therefore serves primarily as evidence of the AMT sample's external validity, and we focus the remainder of our analysis on our AMT participants. (Prior work has already shown that results from AMT are consistent with those of laboratory experiments [23], [13], and that AMT samples "produce reliable results with standard decision-making biases" [11].)

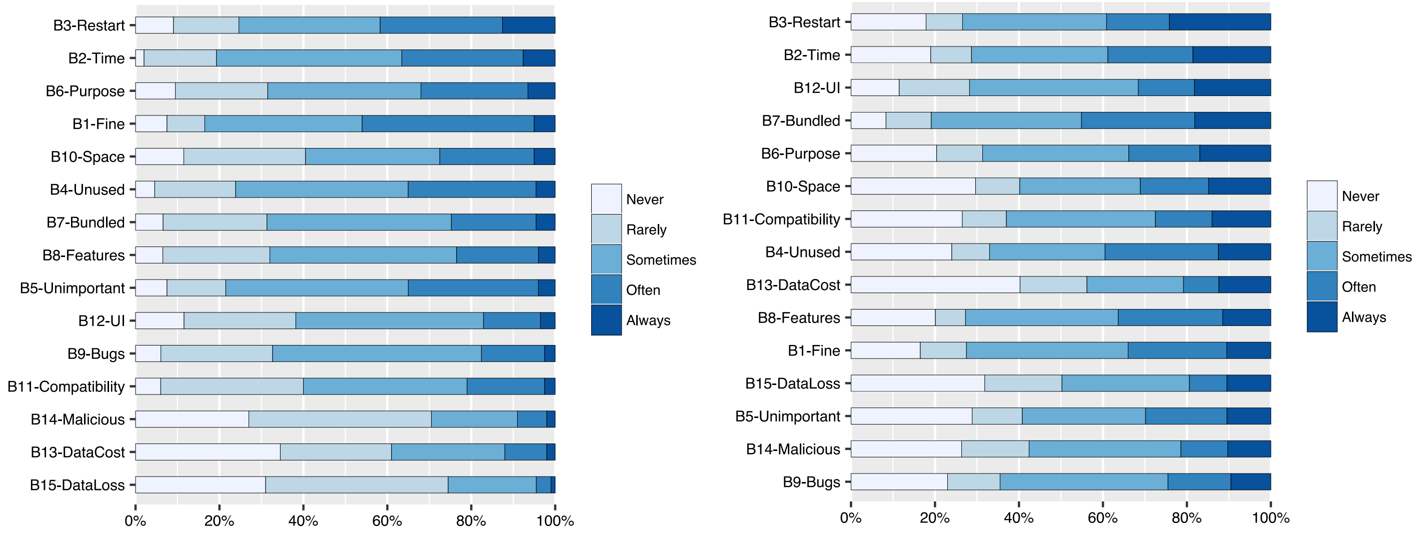
B. *Exploratory Factor Analysis*

In our dataset, we found five eigenvalues greater than 1.0 (the Kaiser criterion [8]), suggesting the presence of up to five factors. We also verified the presence of these five factors using a scree plot (see Figure 2). Together, these five factors explained 62% of the variance in the data. Subsequently, we extracted these five factors using a PCA, applied a Varimax rotation and considered a belief loaded on a factor if its loading exceeded 0.5. We ignored an item if it loaded on a factor, but its loading on that factor was not twice as high as its loading on the other factors [26]. These factor loadings are presented in Table II.

After this step, we ignored beliefs B4, B7, B8, B9, B11, and B12, as they failed to load predominantly on a single factor. We note that this does not reflect the importance of these beliefs, but only that they are not strongly correlated or that they are held at equal rates by the population. This is a necessary step for PCA, providing a more reliable basis to draw conclusions from, as leaving in individual questions would cause too much bias.

Left with the remaining 9 beliefs, we repeated the PCA, Varimax rotation, and computed the new loadings. We extracted three factors explaining 56% of the variance in the data. The output of this step is shown in Table III. We emphasize again that we could not repeat this analysis on the GCS sample since those participants were not the same across the GCS survey questions (i.e., each GCS participant only answered a single question).

The process left us with three distinct factors, each representing 2–4 beliefs. We gave each of the factors names based on the ideas that unify the beliefs that loaded onto them:



(a) Amazon Mechanical Turk

(b) Google Consumer Surveys

Fig. 1: The distribution of software updating beliefs from both the Amazon Mechanical Turk and Google Consumer Surveys samples.

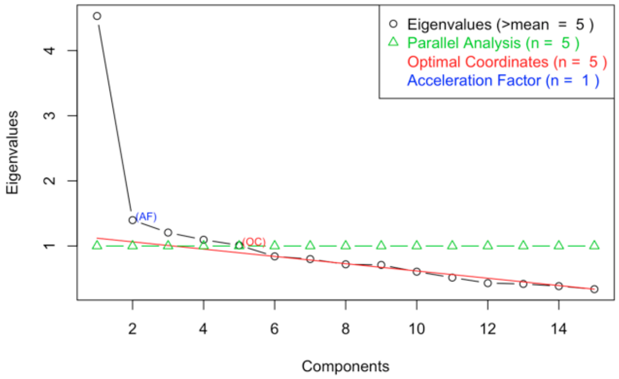


Fig. 2: Scree plot indicating the number of factors we discovered during the Exploratory Factor Analysis.

Belief	**Fac. 1**	**Fac. 2**	**Fac. 3**	**Fac. 4**	**Fac. 5**
B3-Restart	0.651				
B8-Features	0.620		0.363		
B7-Bundled	0.582		0.397		
B2-Time	0.581				
B4-Unused	0.509			0.345	
B9-Bugs	0.455		0.452		
B1-Fine		0.840			
B5-Unimport.		0.794			
B6-Purpose		0.629			
B14-Malicious			0.785		
B15-DataLoss			0.761		
B12-UI	0.407	0.357	0.461		
B13-DataCost				0.866	
B11-Compatib.				0.551	0.524
B10-Space					0.790

TABLE II: Factor loadings from the first PCA with Varimax rotation. Only loadings >0.25 are shown.

- *Update Necessity*: These beliefs concern failing to understand why updates are required and what purpose they serve:
  - B1 ("Updates seem unnecessary as everything works fine")
  - B5 ("Updates seem unimportant")
  - B13 ("Updates consume a lot of data on Internet plans")
- *Update Costs*: These beliefs represent the costs users incur while updating (whether they be time, resources, etc.):
  - B2 ("Updates take a long time to install")
  - B3 ("Updates require unnecessary restarts of applications or PCs")
  - B10 ("Updates occupy a lot of disk space")
- *Update Risks*: These beliefs represent risks faced during or after updating:
  - B14 ("Updates contain malicious software")
  - B15 ("Updates lead to a loss of data")

C. *Prevalence of Beliefs*

We calculated the percentage of AMT participants whose beliefs were described by each of the factors: for the beliefs comprising each factor, we took the average Likert score, and then applied the same threshold we used for PCA. In this manner, we observed that 40.5% of participants held beliefs about *Update Costs*, 29.2% held beliefs about *Update Necessity*, and 7.5% held beliefs about *Update Risks*.

To examine discriminant validity between the three factors, we performed a Spearman correlation. We observed that while the three factors were inter-correlated, these correlations are

Belief	Necessity	Costs	Risks
B1-Fine	0.779		
B5-Unimportant	0.743	0.318	
B13-DataCost	0.546		
B2-Time		0.745	
B3-Restart		0.709	
B10-Space		0.536	
B14-Malicious			0.819
B15-DataLoss			0.810

TABLE III: Factor loadings from the second PCA with Varimax rotation, after removing beliefs that did not load at least twice as highly on a single factor during the previous PCA. Only loadings >0.25 are shown.

not very strong: $r = 0.478$ when comparing *Updating Costs* with *Updating Risks*, $r = 0.487$ when comparing *Updating Necessity* with *Updating Risks*, and $r = 0.597$ when comparing *Updating Necessity* with *Updating Costs*. Thus, participants may have beliefs relating to one factor, but not others.

D. Limitations

A limitation of our study is that we did not follow-up our exploratory factor analysis with a confirmatory one. Future research could replicate our analyses and validate the factors we found. Our results are also generalizable only to the AMT population. However, we supplemented our sample from AMT with another from GCS to provider a comparison of the beliefs. Further, while the AMT population is limited in how diverse it is, research [2], [27] has shown that it is similar to university students and other online participant pools. Future studies may also wish to replicate our results with larger sample sizes. We note, however, that we verified that our participant-to-item ratio (200/15 = 13:1) was within the bounds considered sufficient for exploratory factor analysis as shown by a meta study [6].

V. Discussion

The ultimate goal of any research on software updates is to make users more secure by increasing the number of out-of-date systems that get patched. Our research contributes to this goal by increasing our community's understanding of the reasons why people do not update their systems.

A. Software Updating Beliefs

Our study is the first, to our knowledge, to quantify the frequency with which various beliefs about updating appear in the population. We found that most people expressed agreement with nearly all of the reasons for avoiding software updates uncovered by previous studies, to varying degrees (see Figure 1). Except for narrow concerns (e.g., platform-specific), such as mobile data usage, and ones whose dangers have not been widely advertised (updates containing malicious software), the majority of respondents said each reason was at least sometimes true.

Respondents found certain aspects of the updating process especially annoying. About 40% of the AMT and GCS sample stated that updates either "often" or "always" required unnecessary restarts. Similarly, the duration of the installation process was another frequent complaint, expressed by nearly 40% of both the AMT and the GCS respondents, who said that this was either "often" or "always" a problem.

B. Factors from Beliefs

In addition to the prevalence of individual beliefs, our analysis uncovered several unifying factors. This finding suggests that, rather than holding an assortment of distinct and disconnected opinions, people subscribe to "packages" of beliefs about software updating. These represent three distinct axes along which users' concerns are expressed.

The first is *Update Necessity*. Many respondents were unconvinced of the need for updates, agreeing that they "seem unimportant" because "everything works fine." An orthogonal issue was the toll updates impose on users, which we refer to as *Update Costs*. They take time and disk space, and interrupt users' workflow when they demand a restart. Finally, some worried about *Update Risks*, bad things that may happen if an update goes wrong: lost data or worse.

C. Implications for Software Developers

1) Update Messaging: Our results suggest that one relatively low-cost way of getting users to install updates is by solving the information problem, i.e., accompanying updates with better messages. Currently, software updates typically present the same generic information about themselves to every user. These often lack concrete information and do not directly address users' inherent reservations and beliefs about updating.

The dominant factors identified by our study suggest concrete topics update messaging can address. For example, to address beliefs about *Update Necessity*, developers can clarify the purpose specific updates serve, explaining that they may be important even if everything is working fine. This point is especially critical to make in the case of security updates, which often fix problems that users are unlikely to even be aware of.

As another example, to address beliefs about *Update Costs*, developers can clarify how users might be impacted (e.g., the duration of the installation process) during the time of the update, and what if any, steps they need to take before installing the update for their specific operating system. Conversely, if an update does not require a cost to end-users and runs in the background, the update message can clarify that. Future work could test these messages by means of controlled experiments.

2) Designing and Deploying Updates: Beyond simply messaging, developers should take these factors into account when designing software and updating mechanisms. Thus, as the necessity of restarting is a common complaint, and a major contributor to *Update Costs*, systems that apply updates silently and in the background would be welcomed by users.

Developers can take further steps to reduce their users' *Update Risks*. By using secure and verifiable updating mechanisms, they can reduce the possibility of malicious updates. And by rigorous testing, the risk of data loss can be reduced.

Many modern systems have turned to automatic updates, which has has proven to be an effective and successful system [20]. In cases where this is not an option, it is to the benefit of both parties for software developers to be able to convince their users to install updates. For this to happen, a solid understanding of people's current beliefs, such as that established by this study, is a fundamental prerequisite.

VI. Conclusion

We used a large scale survey to measure the prevalence of different beliefs users have about software updates that were discovered by previous qualitative literature [32], [29], [31], [34], [18], [10]. We found that these beliefs can be grouped into three factors that each represent a different facet of the beliefs: *Update Necessity*, *Update Costs*, *Update Risks*. These factors provide several practical recommendations for how software developers can improve existing update systems, including how and which of these previously discovered beliefs should be targeted and addressed.

VII. Acknowledgements

This research was supported by NSF grant CNS-1528070, BSF grant #2014626 and The Center for Long-Term Cybersecurity at the University of California, Berkeley.

References

[1] I. Ajzen, "The theory of planned behavior," *Organizational behavior and human decision processes*, vol. 50, no. 2, pp. 179–211, 1991.

[2] C. Bartneck, A. Duenser, E. Moltchanova, and K. Zawieska, "Comparing the similarity of responses received from studies in amazon's mechanical turk to studies conducted online and with direct recruitment," *PloS one*, vol. 10, no. 4, p. e0121595, 2015.

[3] M. Buhrmester, T. Kwang, and S. D. Gosling, "Amazon's Mechanical Turk: A New Source of Inexpensive, Yet High-Quality, Data?" *Perspectives on Psychological Science*, vol. 6, no. 1, pp. 3–5, Jan. 2011. [Online]. Available: http://journals.sagepub.com/doi/10.1177/1745691610393980

[4] Cisco, "Cisco Annual Security Report," https://www.cisco.com/web/offers/pdfs/cisco-asr-2015.pdf, 2015.

[5] H. Coolican, *Research methods and statistics in psychology.* Hodder & Stoughton Educational, 1990.

[6] A. B. Costello and J. W. Osborne, "Best practices in exploratory factor analysis: Four recommendations for getting the most from your analysis," *Practical assessment, research & evaluation*, vol. 10, no. 7, pp. 1–9, 2005.

[7] E. Derr, S. Bugiel, S. Fahl, Y. Acar, and M. Backes, "Keep me updated: An empirical study of third-party library updatability on android," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 2187–2200.

[8] L. R. Fabrigar, D. T. Wegener, R. C. MacCallum, and E. J. Strahan, "Evaluating the use of exploratory factor analysis in psychological research." *Psychological methods*, vol. 4, no. 3, p. 272, 1999.

[9] M. Fagan, M. M. H. Khan, and R. Buck, "A Study of Users' Experiences and Beliefs about Software Update Messages," *Computers in Human Behavior*, vol. 51, Part A, pp. 504 – 519, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0747563215003854

[10] A. Forget, S. Pearman, J. Thomas, A. Acquisti, N. Christin, L. F. Cranor, S. Egelman, M. Harbach, and R. Telang, "Do or do not, there is no try: User engagement may not improve security outcomes," in *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*. Denver, CO: USENIX Association, Jun. 2016, pp. 97–111. [Online]. Available: https://www.usenix.org/conference/soups2016/technical-sessions/presentation/forget

[11] J. K. Goodman, C. E. Cryder, and A. Cheema, "Data collection in a flat world: The strengths and weaknesses of mechanical turk samples," *Journal of Behavioral Decision Making*, vol. 26, no. 3, pp. 213–224, 2013. [Online]. Available: http://dx.doi.org/10.1002/bdm.1753

[12] J. He and F. J. van de Vijver, "Self-presentation styles in self-reports: Linking the general factors of response styles, personality traits, and values in a longitudinal study," *Personality and Individual Differences*, vol. 81, no. Supplement C, pp. 129 – 134, 2015, dr. Sybil Eysenck Young Researcher Award. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S019188691400511X

[13] J. J. Horton, D. G. Rand, and R. J. Zeckhauser, "The online laboratory: Conducting experiments in a real labor market," *Experimental Economics*, vol. 14, no. 3, pp. 399–425, 2011.

[14] H. Hotelling, "Analysis of a complex of statistical variables into principal components." *Journal of educational psychology*, vol. 24, no. 6, p. 417, 1933.

[15] I. Ion, R. Reeder, and S. Consolvo, "““...No one Can Hack My Mind": Comparing Expert and Non-Expert Security Practices," in *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*. Ottawa: USENIX Association, Jul. 2015, pp. 327–346. [Online]. Available: https://www.usenix.org/conference/soups2015/proceedings/presentation/ion

[16] F. Leutner, G. Ahmetoglu, R. Akhtar, and T. Chamorro-Premuzic, "The relationship between the entrepreneurial personality and the big five personality traits," *Personality and Individual Differences*, vol. 63, no. Supplement C, pp. 58 – 63, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0191886914000622

[17] A. Mathur and M. Chetty, "Impact of user characteristics on attitudes towards automatic mobile application updates," in *Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017)*. Santa Clara, CA: USENIX Association, 2017, pp. 175–193. [Online]. Available: https://www.usenix.org/conference/soups2017/technical-sessions/presentation/mathur

[18] A. Mathur, J. Engel, S. Sobti, V. Chang, and M. Chetty, "””they keep coming back like zombies": Improving software updating interfaces," in *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*. Denver, CO: USENIX Association, Jun. 2016, pp. 43–58. [Online]. Available: https://www.usenix.org/conference/soups2016/technical-sessions/presentation/mathur

[19] Microsoft, "Microsoft Security Intelligence Report Volume 20: July through December 2015," http://bit.ly/2BF32Vp, 2015.

[20] A. Nappa, R. Johnson, L. Bilge, J. Caballero, and T. Dumitras, "The attack of the clones: A study of the impact of shared code on vulnerability patching," in *Proceedings of the 2015 IEEE Symposium on Security and Privacy*, ser. SP '15. Washington, DC, USA: IEEE Computer Society, 2015, pp. 692–708. [Online]. Available: http://dx.doi.org/10.1109/SP.2015.48

[21] L. H. Newman, "Equifax officially has no excuse," Wired, September 14 2017, https://www.wired.com/story/equifax-breach-no-excuse/.

[22] M. Oltrogge, Y. Acar, S. Dechand, M. Smith, and S. Fahl, "To pin or not to pin helping app developers bullet proof their tls connections," in *Proceedings of the 24th USENIX Conference on Security Symposium*, ser. SEC'15. Berkeley, CA, USA: USENIX Association, 2015, pp. 239–254. [Online]. Available: http://dl.acm.org/citation.cfm?id=2831143.2831159

[23] G. Paolacci, J. Chandler, and P. G. Ipeirotis, "Running experiments on amazon mechanical turk," *Judgment and Decision Making*, vol. 5, no. 5, 2010.

[24] E. Peer, J. Vosgerau, and A. Acquisti, "Reputation as a sufficient condition for data quality on amazon mechanical turk," *Behavior research methods*, vol. 46, no. 4, pp. 1023–1031, 2014.

[25] E. Pettersson, J. Mendle, E. Turkheimer, E. E. Horn, D. C. Ford, L. J. Simms, and L. A. Clark, "Do maladaptive behaviors exist at one or both ends of personality traits?" *Psychological assessment*, vol. 26, no. 2, p. 433, 2014.

[26] G. Saucier, "Mini-markers: A brief version of Goldberg's unipolar Big-Five markers," *Journal of personality assessment*, vol. 63, no. 3, pp. 506–516, 1994.

[27] D. J. Simons and C. F. Chabris, "Common (mis) beliefs about memory: A replication and comparison of telephone and mechanical turk survey methods," *PloS one*, vol. 7, no. 12, p. e51876, 2012.

[28] H. Suh, N. Shahriaree, E. B. Hekler, and J. A. Kientz, "Developing and validating the user burden scale: A tool for assessing user burden in computing systems," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, ser. CHI '16. New York, NY, USA: ACM, 2016, pp. 3988–3999. [Online]. Available: http://doi.acm.org/10.1145/2858036.2858448

[29] Y. Tian, B. Liu, W. Dai, B. Ur, P. Tague, and L. F. Cranor, "Supporting Privacy-Conscious App Update Decisions with User Reviews," in *Proceedings of the 5th Annual ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices*, ser. SPSM '15. New York, NY, USA: ACM, 2015, pp. 51–61. [Online]. Available: http://doi.acm.org/10.1145/2808117.2808124

[30] United States Computer Emergency Readiness Team, "Before You Connect a New Computer to the Internet," https://www.us-cert.gov/ncas/tips/ST15-003, December 2015.

[31] K. Vaniea and Y. Rashidi, "Tales of software updates: The process of updating software," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, ser. CHI '16. New York, NY, USA: ACM, 2016, pp. 3215–3226. [Online]. Available: http://doi.acm.org/10.1145/2858036.2858303

[32] K. E. Vaniea, E. Rader, and R. Wash, "Betrayed by updates: How negative experiences affect future security," in *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*, ser. CHI '14. New York, NY, USA: ACM, 2014, pp. 2671–2674. [Online]. Available: http://doi.acm.org/10.1145/2556288.2557275

[33] R. Wash and E. Rader, "Too Much Knowledge? Security Beliefs and Protective Behaviors Among United States Internet Users," in *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*. Ottawa: USENIX Association, Jul. 2015, pp. 309–325.

[34] R. Wash, E. Rader, K. Vaniea, and M. Rizor, "Out of the Loop: How Automated Software Updates Cause Unintended Security Consequences." USENIX Association, 2014, pp. 89–104.