What risk? I don't understand. An Empirical Study on Users' Understanding of the Terms Used in Security Texts

Tingmin Wu
tingminwu@swin.edu.au
Swinburne University of Technology
Data61, CSIRO

Rongjunchen Zhang
rongjunchenzhang@swin.edu.au
Swinburne University of Technology
Data61, CSIRO

Wanlun Ma
mawanlun@std.uestc.edu.cn
University of Electronic Science and Technology of China

Sheng Wen*
swen@swin.edu.au
Swinburne University of Technology

Xin Xia
xin.xia@monash.edu
Monash University

Cecile Paris
Cecile.Paris@data61.csiro.au
Data61, CSIRO

Surya Nepal
Surya.Nepal@data61.csiro.au
Data61, CSIRO

Yang Xiang
yxiang@swin.edu.au
Swinburne University of Technology

ABSTRACT

Users receive a multitude of security information in written articles, e.g., newspapers, security blogs, and training materials. However, prior research suggests that these delivery methods, including security awareness campaigns, mostly fail to increase people's knowledge about cyber threats. It seems that users find such information challenging to absorb and understand. Yet, to raise users' security awareness and understanding, it is essential to ensure the users comprehend the provided information so that they can apply the advice it contains in practice.

We conducted a subjective study to measure the level of users' understanding of security texts. We find that 61% of the terms security experts used in their writings are hard for the public to understand, even for people with some IT backgrounds. We also observe that 88% of security texts have at least one such term. Moreover, we notice that existing dictionaries, including the online ones (e.g., Google Dictionary), cover no more than 35% of the terms found in security texts. To improve users' ability to understand security texts, we developed a framework to build a user-oriented security-centric dictionary from multiple sources. To evaluate the effectiveness of the dictionary, we developed a tool as a service to detect technical terms and explain their meanings to the user in pop-ups. The results of a subjective study to measure the tool's performance showed that it could increase users' ability to understand security articles by 30%.

*Corresponding author.

ASIA CCS '20, October 5–9, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-6750-9/20/10…$15.00
https://doi.org/10.1145/3320269.3384761

CCS CONCEPTS

• **Security and privacy → Usability in security and privacy**; • **Human-centered computing → User interface toolkits**.

KEYWORDS

user security awareness; security term explanation; user study

ACM Reference Format:
Tingmin Wu, Rongjunchen Zhang, Wanlun Ma, Sheng Wen, Xin Xia, Cecile Paris, Surya Nepal, and Yang Xiang. 2020. What risk? I don't understand. An Empirical Study on Users' Understanding of the Terms Used in Security Texts. In *Proceedings of the 15th ACM Asia Conference on Computer and Communications Security (ASIA CCS '20), October 5–9, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3320269.3384761

1 INTRODUCTION

In our increasingly digitised and interconnected society, cyber attacks continue to escalate and harm internet users [23]. It is recognised that humans are still the dominant security decision-makers in the face of cyber attacks [15]. In 2017, Netwrix conducted a survey designed to identify IT security, compliance, and operational risks that organisations around the globe face on a daily basis. In that survey, all government entities considered their employees as the biggest threat [11]. Education in understanding security texts is critical to the improvement of users' ability in making correct security decisions [42]. However, it was revealed that less than 25% of security advice was easy to understand [16].

The fact is that most users are not security experts, even if they are technically savvy. User studies conducted through interviews revealed that two-third of the users underestimate the extent of cyber harms, and only around 10% can explain protective measures (e.g., fraud alerts) correctly [64]. Despite efforts to increase users' understanding of security measures such as removing terms [18] and improving security interfaces [1], the low success rate shows that it is still challenging to get users to the stage that they can apply security measures in practice [17]. The critical step in achieving this goal is to help users better understand security terms.

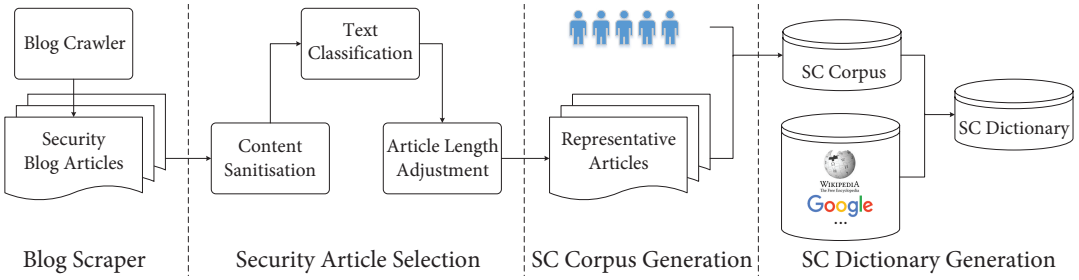


Figure 1: Overview of our framework to build the Security-Centric (SC) Dictionary.

Research conducted by the Pew Research Centre tested users' knowledge about technical terms [10]. It indicated that most users were familiar with common terms (e.g., 'wiki'), but had difficulty in explaining certain concepts (e.g., 'phishing', 'virus'). Existing studies also revealed that security advice to reduce risks [44] is helpful, but requires a high level of education to understand and is likely to be misinterpreted or even ignored [13, 51]. Other studies report complementary facts like half of users refuse to use security advice because they think the concerns are unnecessary or fail to comprehend how it works [29, 45, 61].

Despite the reported problem in these studies, there has been no analysis of why the comprehension level of users is low when it comes to security advice, and what steps can be taken to alleviate this issue. We focus on text-based means of delivering security messages (i.e., security texts) and try to address these questions by a series of real-world experiments, designed to answer the four following research questions.

RQ1. What are the technical terms used in security texts and their difficulty levels from a user' perspective?

RQ2. Are traditional methods useful in measuring the difficulty level of technical terms?

RQ3. Are the technical terms as difficult for people with IT background as they are for those without IT background?

RQ4. What functions would users like to help them read technical articles?

Our first step was to build a dataset of security blogs, since they are one of the major public sources providing news and articles containing computer security advice [36]. A study revealed that most users learn cybersecurity through media, especially blogs [51]. We then invited 597 participants to take part in a subjective study, in which we asked them to rate their comprehension of these blogs. We generated a security corpus consisting of 6,286 technical terms from the results. The study also reveals interesting and surprising results, some of which are reported below:

- 61% of technical terms are considered to be hard (with a difficulty level higher than 50% on a scale from 1 to 10) and have a serious impact on the comprehension of security texts;
- There exists an inconsistency between users' reported difficulty levels and those of traditional readability tests, e.g., the termhood calculation [12] and term occurrences in Google Search;
- People with IT background assign higher difficulty levels to the technical terms related to cyber threats and protection measures compared to those without IT background; and
- 65% of participants would like to have a dictionary-based explanation for technical terms.

The last finding, which was the result of analysing the answers to an open question in our survey, motivated us to perform an additional study, mainly to test ways to improve security text readability from a user' perspective. General dictionaries (Wikipedia Page Previews [59], Google Dictionary [22] and Mac Dictionary [4]) were not useful, because none of them was able to cover more than 35% of the collected terms. Hence, we built a specific dictionary by combining multiple sources (cf Section 4). Fig.1 visually shows the steps taken for this purpose. We then developed a service tool as a browser plug-in. This service used our dictionary to provide explanations for security terms in the form of pop-ups (cf Section 4).

To find out to what extent our tool helps users understand security texts and to see what influences users' comprehension, we conducted a second experiment. We employed 112 participants with different IT backgrounds to explore the factors that influence their understanding. The analysis revealed the following:

- Our tool can help users understand security articles significantly better, as much as 30%, than existing methods.
- Users misunderstand ambiguous terms (e.g., terms with multiple meanings or with meanings similar to other terms).
- Users with IT background perform better in understanding security texts than those without, but only when using our tool.

We believe these findings can help security experts compose their security advice in high readability with users in mind, and also develop tools and methods for a more effective delivery of security texts. To summarise, our paper has the following three contributions:

- We conducted an empirical study to understand the difficulties faced by users in comprehending security texts.
- We built a user-oriented security-centric dictionary.
- We developed and implemented a tool as a service by using our dictionary to help users comprehend security texts.

The rest of the paper is structured as follows. In Section 2, we review related work. Sections 3 and 4 report the details of our experiments and their results. Section 5 discusses the implications and limitations. Section 6 concludes the paper.

2 RELATED WORK

In this section, we discuss related work on users' perceptions of security risks, their understanding of security threats, measures and descriptions, and security education.

Perception of Security Risks. Users' awareness of security threats is a great concern for computer security experts. Fagan and Khan investigated the difference in risk perception between those who followed the security advice and those who did not [17]. Security advice was reported to be incomprehensible to some home computer users who lacked any high education in [13]. Therefore, these users were unable to take appropriate actions to counter security threats.

Wash conducted qualitative interviews to find out how well home computer users understood security threats [60]. He also identified eight folk models of security threats, including malware and attackers, which exposed users' misunderstanding of the concepts and explained why they ignored security warnings. Routi et al. conducted a series of interviews to investigate users' perceptions of online security [51]. They found that users' misunderstandings of browser-based TLS (Transport Layer Security) indicators caused unsafe behaviours. Wash and Rader also found that participants had different security knowledge and beliefs about viruses and hackers [61].

In the Android ecosystem, Harbach et al. generated personalised examples to improve users' awareness when making security and privacy decisions, e.g., during the app installation process [25]. Similarly, some studies [62, 63] applied static code analysis and generated security-centric descriptions or privacy policies with different sentence structures.

Security Understanding. Howe et al. made a literature review on existing surveys and found that, although users were aware of and concerned about security threats, they were unable to understand them [27]. Similarly, Shay et al. interviewed 394 people about the hijacking problem [55]. The results reflected that the users were aware of malware, phishing, and third-party breaches but unable to apply adequate security measures. Ion et al. demonstrated the difference of security advice between experts and non-experts, such as using 2FA and password manager to prevent attacks, compared to using anti-virus programs and changing passwords frequently [29]. Later, Zou et al. built mental models of credit bureaus and found the participants were aware of the data breaches, but they hardly understood them and thus suffered from them [64]. The reason could be that they underestimate the possibility to become victims, so that they refused to take effective measures in time.

The factors that influence users' understanding of computer security were also studied in some recent work. Forget et al. presented the relationships between users' attitudes or behaviours and their understanding of security threats [21]. The bias on the estimation of their technical expertise and misunderstanding of the risks could enable severe attacks by applying wrong security measures. Acquisti et al. explained what affected users in security and privacy decision making [2]. Sawaya et al. also found that users' self-confidence knowledge affected their security behaviours more than actual knowledge [53].

In addition to users' knowledge, the content of security texts (e.g., newspapers, security blogs) can also affect users' understanding. Badal et al. revealed that one critical reason that the users did not apply appropriate security measures was the poor design of security descriptions [7]. Redmiles et al. found users with more enterprise knowledge were more likely to take the socioeconomically advantage to obtain security advice from the workplace, while low-skill people mostly learned from their peers [44].

Users' Security Education. Dale et al. highlighted the necessity of cybersecurity education for raising users' awareness [50]. Great efforts have been made to improve the delivery of security texts to end users, with the purpose of education. For example, SSL (Secure Sockets Layer) warnings were re-written in simple language by removing technical terms for browser users to understand them [18]. Similarly, privacy notices [54], warning habituation, and security interventions [3, 52] were studied to help users make correct security decisions.

There are also some studies about educational approaches or tool development to protect users from phishing attacks [5, 6, 30]. Other significant work suggested the users should select stronger passwords and store them safely [20, 38, 57, 58]. Some researchers emphasised the importance of the improvement of security tools and interfaces [1, 35].

Existing works mainly revealed the fact that users' risk perception and security behaviours will be affected if users do not follow security advice because they do not understand the texts. Compared to the previous works, our research focuses on investigating users' difficulty in understanding security texts. Moreover, we make efforts in explaining the terms that users cannot understand instead of removing them or rewriting the sentences. Our work aims at helping users make the right security decisions based on their knowledge.

3 EXPERIMENT 1: USERS' UNDERSTANDING OF SECURITY TEXTS

We conducted a study with 597 participants to learn users' comprehension of security risks and to answer the three research questions presented in the introduction. Both this study and evaluation study (Section 4.2) were approved by the CSIRO Social and Interdisciplinary Human Research Ethics Committee[1].

3.1 Setup and Methodology

3.1.1 Data Sources. We used the blogs about cybersecurity as our source. They were selected based on the rankings on recommendation websites and popularity in social media. We also included the technical blogs from a previous study [32] such as TrendMicro[2]. Those blogs publish news, articles and technical reports analysis about the latest trends in cybersecurity for different types of audience and not only for security experts. For example, The Hacker News[3] attempts to educate users with varying technology backgrounds to stay safe online, and it has attracted over 2 million Facebook followers.

We implemented a crawler in Python using its Beautiful Soup [48] library. The crawler first scraped all page links from the homepage and extracted HTML pages from these links. The crawler then scraped the links of technical articles, which were stored in tags, and attributes of the extracted HTML pages. Regular expressions were applied to avoid unrelated content, e.g., advertisements, blog

[1] Ethics Clearance 172/19
[2] https://blog.trendmicro.com/
[3] https://thehackernews.com/

contributors' biographies, and outdated articles. For example, we applied "`((?!:).)*/201[5-8]/)(\d2/)((?!:).)*(.html)$`" to extract the blogs from 2015 to 2018. The selected articles were then extracted and downloaded as HTML files. Our collection was conducted in December 2018, and only the latest articles (starting from 2015) were collected. In total, we collected 42,409 cybersecurity articles from 35 technical blogs (see supplementary table[4]). In the next step, a web-based questionnaire was designed for users to annotate technical terms and their difficulty levels.

3.1.2 Data Pre-processing. A user study on over 40,000 articles is challenging and time-consuming. Therefore, through data pre-processing, we chose some representative articles with which to conduct the study. Generally, the reading speed of an adult is around 275 words per minute [37]. We removed the articles which required less than one minute to read (19.2% of the articles). We further removed the long ones that may take more than five minutes to read.

We used the topic modelling technique [9] to select the representative articles. Topic modelling is a statistical method for discovering the abstract "topics" that appear in a collection of documents. We used LDA (Latent Dirichlet Allocation) [9], a state-of-the-art method for topic modelling. LDA is an unsupervised learning algorithm which discovers a mixture of different topics for each document with distinguished probabilities. We implemented LDA in Python using gensim [46]. Coherence measures were employed to evaluate the performance of the generated topic models as they have better human interpretability than other measures such as perplexity [49]. We applied the module `CoherenceModel` in gensim to obtain LDA models as well as their topic coherence. To determine the number of topics, we compared the coherence of generated models with different values (i.e., 5, 10, 15, ..., 50), and kept other parameters as default. In our experiments, we noted that the highest coherence score was at 10. Therefore, we set LDA to discover ten topics. We then manually interpreted the discovered topics. The topics are government/company reports, device/system access, vulnerability/flaw, file/code, user account security, network attacks, data breaches, security threat/cyber risks in business, malicious software (e.g., malware, ransomware), and non-technical news. These topics covered all articles. We randomly picked 20 articles from each topic. In total, 200 representative articles were chosen for the study.

An existing study [42] identified ten topics of security advice, as shown in Fig.2. We find these ten topics are contained in our LDA-generated topics at different granularity. The numbers of the articles in our dataset for all the identified topics are depicted in Fig.2. For almost all the topics, we find at least 10% of articles in our dataset are related. It indicates we have sufficient coverage. Besides, our dataset contains advanced security articles for professionals, such as complex operations to manually remove malware and detailed attacking methods with file operations or commands.

3.1.3 Study Methodology and Procedure. We designed a questionnaire to measure users' understanding of the articles. We were mainly after the terms they found difficult to understand. The questionnaire contained three parts: questions about demographics, an annotation task, and questions about the articles. Instructions and a tutorial were provided at the beginning of the questionnaire. We also provided some example terms along with their difficulty levels which were determined by some readability calculation methods.

[4]https://ktd4869.github.io/Reading_Test_MT/supplementary_table.pdf

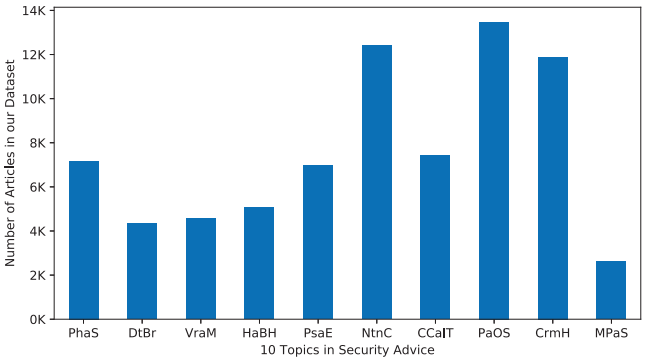


Figure 2: The number of the articles in the identified ten topics of security advice [42]: Phishing and Spam (PhaS), Data Breaches (DtBr), Viruses and Malware (VraM), Hackers and Being Hacked (HaBH), Passwords and Encryption (PsaE), National Cybersecurity (NtnC), Credit Card and Identity Theft (CCaIT), Privacy and Online Safety (PaOS), Criminal Hacking (CrmH), and Mobile Privacy and Security (MPaS).

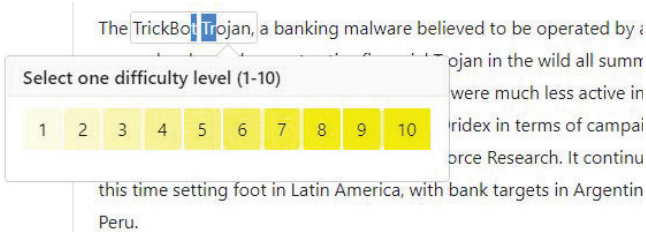


Figure 3: A screenshot of our technical term annotation tool.

We employed Amazon Mechanical Turk (MTurk) to conduct our study. MTurk is a marketplace where individuals can outsource tasks with monetary compensation. We published the questionnaire with 2 U.S. dollars (rewards) for each completion. The workers were required to be 18 years or older and proficient in English reading/writing to participate in the user study. Only the workers with a 95% approval rating (suggested in [40]) were eligible to participate in our survey.

In the demographics section, participants were asked about their gender, age, education, IT background, whether they were English native speaker and four questions about their experience on security threats. They were allowed to choose 'prefer not to answer'.

Workers were then required to annotate the articles through our designed interface, as shown in Fig.3. Each time participants clicked and selected a term (or a phrase up to five words), they were asked to choose its difficulty level in a pop-up window. The difficulty scale was from 1 to 10. Once chosen, the term would be highlighted with a yellow tone whose brightness showed the level of difficulty. The annotation function was implemented based on [34].

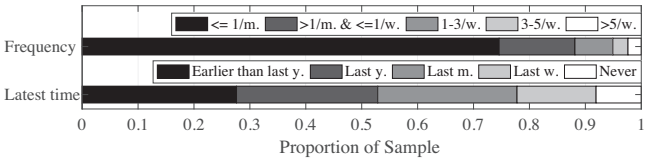


Figure 4: Users' experience about security threats (top: frequency (times/week, month, or year); bottom: time of last experience).

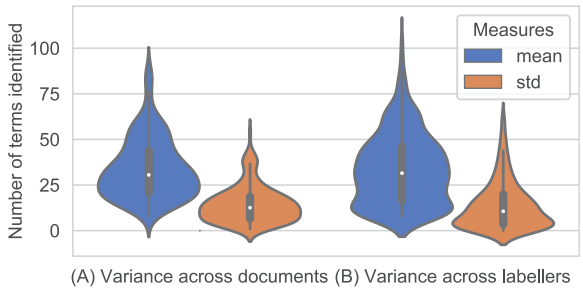


Figure 5: The overall distribution of *mean* and *std* in the number of terms identified by (A) different labellers across 200 documents and (B) different documents across 597 labellers.

After annotating the articles, workers were asked some questions about the article. More specifically, they were asked to select two of the terms they had highlighted and to explain their choice of difficulty. They were also asked to answer two open questions and describe the desired functionalities from a tool that they thought could help them understand the terms.

In total, we collected 597 valid responses after a manual review on the submitted assignments (*mean*: 25.67 min; *std*: 11.82 min). We rejected unsatisfactory assignments such as random or blank answers and careless term annotation (mostly annotated words not related to computer science or with less than three annotations). Each participant was allocated two articles, selected randomly from our 200-article pool, and each of the 200 articles was annotated by at least three participants.

3.1.4 Data Analysis. We analysed the responses to explore users' comprehension of security texts to answer our research questions. We used open card sorting [56] to group the answers to the open questions. We extracted the keywords from each answer and grouped the answers by matching the keywords. The details are described in the next sub-section.

3.2 Survey Results

3.2.1 Demographics. Our participants are mainly younger adults (70% with ages 18 to 35), and 82% of all the participants are English native speakers. We have an almost equal number of each gender. 81% of younger adults have bachelor or higher degrees. This percentage only slightly decreases to 71% for older adults. 50% of the younger adults have IT background, compared to 12% of older adults (ages >50). These statistics show that older adults have less IT knowledge and might be at higher risk against security threats, even with the same education level.

Most of our participants (96%) are daily internet users, but only 8% had never experienced security threats. The frequency and latest experience of security threats are depicted in Fig.4. Most users experienced malware or virus less than once a week, but 40% had an experience at least once a month.

3.2.2 Annotated Term Analysis. We collected the technical terms annotated by MTurk users. Overall, 7,375 terms were collected. We then manually removed invalid terms (e.g., meaningless). Meaningless terms referred to the terms which did not have specific meanings in the IT domain. For example, 'public' is generic and thus removed, but the 'public key' phrase is commonly used in cryptography and was kept. We removed the duplicates of each term and took the average value to replace its difficulty level. Duplicate terms occurred after lemmatisation (e.g., 'APIs' was the plural form of 'API'), punctuation removal, and the removal of the words that did not have a semantic contribution (e.g., 'DNS-based' and 'inject or' were duplicates of 'DNS' and 'inject' respectively). In total, we obtained a 6,286-term security-centric corpus (SC Corpus) which contained 3,276 phrases (e.g., 'web browser', 'ad hoc attacks', 'XSS flaw') and 3,010 words (e.g., '2FA', 'WannaCry', 'malware').

***A.* Data Validity**

Before doing this task, we first conducted a data validity analysis to ensure workers' term annotations were satisfactory. We calculated the *mean* and *std* in the number of terms identified by different labellers within a given document. The distribution of the analysis for all the 200 documents is shown in Fig.5(A). Most of the *std*s range from 6 to 18, with an average of 12. We further analysed the number of terms identified in different documents by a labeller. Fig.5(B) presents the distribution for all 597 labellers. It also indicates relatively low *std*s (mostly <18). Based on this analysis, we conclude that the annotations are of good quality.

We analysed the validity of the difficulty levels labelled by different workers. We calculated the *mean* and *std* in difficulty levels for each term collected. The distribution result for our whole corpus shows the *mean* ranges from 3.5 to 6.7, with an average of 5. The *std*s are mostly from 1.4 to 2.8, with an average of 2.1. We conclude that the resultant difficulty levels are of good quality because they are not significantly different across different labellers.

Term Frequency Distribution Test. Before we answer the research questions, we measure the representativeness of the sampled dataset. We applied the statistical significance test to compare the distributions of the term frequencies in the sampled dataset (200 articles) and the full dataset.

We applied the Log-likelihood ratio test, since it is a core method in corpus comparison [43]. We conducted the analysis for each term in our SC Corpus based on its occurrences in both datasets. The results show that 71.1% of the terms in the SC Corpus have a similar frequency in the full dataset ($p > 0.01$). Therefore, we conclude that our sampled dataset is representative of the full dataset.

***B.* Technical Term Analysis and Results**

We then present the results of the first experiment to answer our four research questions.

RQ1. What are the technical terms used in the security texts and their difficulty levels from a user' perspective?

To explore them, we first identified categories for the terms and evaluated their validity with an expert review. We then analysed the difficulty levels of the terms in each category and the potential impact factor (in which year a term was coined).

Term Categories. We identified the categories of the security terms in the corpus according to their lexical semantics, using the open card sorting [56] again. More specifically, we randomly selected 100 terms from the corpus and identified their categories, and then applied the categories or created new categories for the remaining terms.

We recruited three researchers with the cybersecurity background to complete the manual classification. We used majority voting [39] to identify the categories for each term. If all the researchers had different opinions on a term, then a discussion was conducted until two reached an agreement. Each term was allowed to have up to two categories. The classification result shows that 328 out of 6,286 terms are assigned to two categories, while the rest only have one category. In total, 40 subcategories are generated to classify the corpus. We further consolidated the 40 subcategories into 15 categories. Table 2 in Appendix B lists all the categories and two examples for each subcategory as well as the detailed descriptions.

Expert Review. To ensure the validity of the categories, we conducted an expert interview to evaluate the accuracy of the classifications. We recruited two cybersecurity experts who worked in CSIRO's Data61 for more than two years. Our researchers introduced the categories in details at the beginning. We generated two samples separately, randomly selected as 5% of the corpus, both containing 252 terms. Each reviewer was provided with a sample and asked to highlight the terms not matching the categories. We applied a think-aloud protocol as used in [31]. During the evaluation process, the participants were free to talk about the task, and our researchers were sitting next to them, taking notes and dispelling the doubts. Our results show that both experts thought that only 1 out of 252 terms was labelled incorrectly, which meant our accuracy rate was 99.6%.

Analysis of Categories. We analysed the relationship between labelled difficulty levels and IT background using the chi-squared (χ^2) test. We only kept the terms annotated by both IT and non-IT groups for background comparison, representing 53.47% of the whole corpus. For each term, we calculated the *mean* values of difficulty levels for IT and non-IT groups separately. We find the ratios of harder terms ($difficulty > 5$) in different categories are not related to IT background ($\chi^2 = 0.101, p > 0.999$). The *means* of difficulty levels in different categories are also not related to IT background ($\chi^2 = 0.285, p > 0.999$).

Fig.6 depicts the proportions of different categories in our SC Corpus annotated by all the workers, regardless of their IT background. Each category is further divided into two parts: terms that are hard to understand ($difficulty > 5$) and terms that are more easily understood ($difficulty <= 5$). We find that general terms (e.g., '3D', 'address') form a large part of the corpus, around 32%, and they have relatively low difficulty levels. In contrast, the terms related to protocol/standard (e.g., 'SSL': Secure Sockets Layer), cryptography/authentication (e.g., 'CISM': Certified Information Security Manager) or algorithms (e.g., 'Grovers algorithm') are rarer but are considered as most difficult, with more than 65% terms that are harder understood in each category.

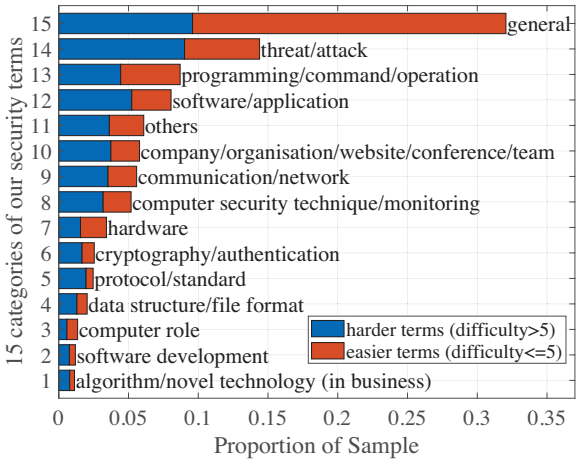


Figure 6: The proportions of harder terms (blue) and easier terms (red) in 15 categories of the SC Corpus.

We investigated the "age" of the terms, that is how long have the terms been used in language. To do this, we searched for the years the terms were coined in different categories of the SC Corpus. Three researchers did this work, and majority voting [39] was used to determine the year. The researchers were required to search each term online to infer the year based on the context of the term. For example, the term 'worm' is ambiguous, but it mostly represents malware computer program in our security articles, so the year the term was coined is when it was first used to describe malware instead of animal 'worm'. Appendix A.1 explains the empirical CDFs (Cumulative Distribution Function) for 7 of our categories.

We analysed the correlation between the years the security terms were coined and the difficulty levels annotated by people with different IT backgrounds in various categories. We calculated the Pearson Correlation Coefficient (Pearson's r) to measure the strength of the correlation between the two variables. Pearson's r ranges from -1 to $+1$. A value of 0 means that there is no correlation. The results show that people with IT background find it harder to understand the newer terms related to programming, software development and threat/attack ($r = 0.32, 0.47, 0.33$ respectively). People without IT background have greater difficulty with recently coined terms in hardware and computer role ($r = 0.34, 0.52$ separately), while old algorithms and technologies are easier for them to understand.

RQ2. Are traditional methods useful in measuring difficulty levels of technical terms?

We explored the difference between users' understanding levels of security concepts and existing termhood measures. We compared the difficulty levels annotated by crowdworkers to the termhood calculated by the traditional measures [12, 19, 26] and the measures based on their occurrences in Google Search. Those measures calculated the terminological degrees based on the term frequencies relative to their frequencies in a general language corpus. WR (weirdness ratio) [12] was applied to compute the termhood for our

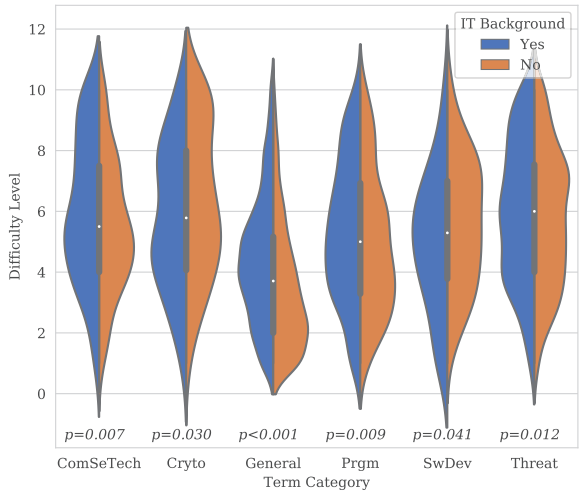


Figure 7: Distribution of difficulty levels in different categories. The figure only includes the six categories with a significant difference ($p < 0.05$) between difficulty levels annotated by people with and without IT background. They are computer security technique/monitoring (ComSeTech), cryptography/authentication (Cryto), general, programming/command/operation (Prgm), software development (SwDev), and threat/attack (Threat).

comparison, which represents the quotient of relative frequency in our corpus and a general language corpus (CLEF 2004 [41]). We also developed a crawler to retrieve the occurrences of our technical terms in Google. A higher amount of Google search queries indicates a higher probability that people have seen these words, which corresponds to lower difficulty levels.

We further tested the significance of the difference between the annotated difficulty levels and these two measures (traditional termhood and occurrences in Google search), respectively. As our results had different scales and correlations from the two measures, we normalised the values to aid comparison by z-score standardisation [14]. We inversely transformed the occurrences in Google search before standardisation, since they correlated negatively to our results. The $p - values$ (< 0.05) of a Mann-Whitney U test indicate both measures are significantly different from our annotated results. Pearson's r (without transformation) shows a similar trend, and the values are -0.017 (our result vs. traditional termhood) and -0.223 (our result vs. occurrences in Google search).

RQ3. Are the technical terms as difficult for people with IT background as they are for those without IT background?

We reviewed the collected terms and their difficulty levels annotated by people with and without IT background separately to investigate the differences among their comprehensions. As analysed in RQ1, the difficulties of the categories in our SC Corpus are unrelated to workers' IT background.

We further analysed the difference between the difficulties of the terms annotated by people with and without IT background in each category. We only kept the terms annotated by both groups for background comparison. We calculated the average difficulty level for each term in each group respectively. For each category, we conducted a Mann-Whitney U test to measure the difference in average difficulty levels between people with different IT backgrounds.

As shown in Fig.7, only 6 out of 15 categories show a statistical difference in difficulty levels annotated by people with and without IT background. The terms in the general category are considered to be the easiest by both groups. It is more challenging for people without IT background to comprehend technical terms in some specific fields (e.g., cryptography/authentication, software development). These particular terms may require further training or education to understand the concepts they represent. We observe that people without IT background are more likely to give lower difficulty levels for the terms they are familiar with, to distinguish them from harder terms, compared to people with IT background. Technically savvy people labelled the terms related to programming as having a higher difficulty level, potentially because they experienced difficulty in applying the related technique in their work. To our surprise, the terms in computer security techniques (measures) and threat/attack are considered harder by people with IT background than those without IT background. This means that having an IT background does not help people understand cybersecurity concepts, and people with IT background are not at lower security risk than those without IT background. This needs further invigoration.

RQ4. What functions would users like to help them read technical articles?

To explore what functions can provide reading assistance, we used the open card sorting [56] again to classify their comments (597 responses) from our survey. Three researchers did the classification with majority voting [39]. Fig.12(A) in Appendix B depicts the proportions of different functions suggested by our participants, where the majority (65%) of people would like to use a dictionary-based tool. 27% of the users felt it was not necessary to have reading assistance. A few participants also raised the need for difficult term detection or highlight in articles and audio assistance to read or pronounce some particular words.

As most users would like a tool to provide definitions or descriptions for the technical terms (i.e., a dictionary-based tool), we further analysed the different methods users mentioned to provide explanations. As shown in Fig.12(B) in Appendix B, the vast majority (81.2%) of the users explicitly described their preference in functions of pop-ups (to hover and define difficult terms) and the dictionary (to provide definitions for lookup). Others also suggested the use of hyperlinks, such as external links to a Wikipedia page or detailed explanations including textual descriptions, videos, and graphics. Similar to the dictionary, a glossary of various acronyms and jargons was also suggested.

We list some representative comments below:

- *"I would like to have a tool to help read an article like this. The ideal features I would look for will be a feature like Kindle's dictionary. If I long-press or hover over the word, there should be brief info about the word."*
- *"A built-in browser tool that defines terms or links to the context within annotations might be useful."*

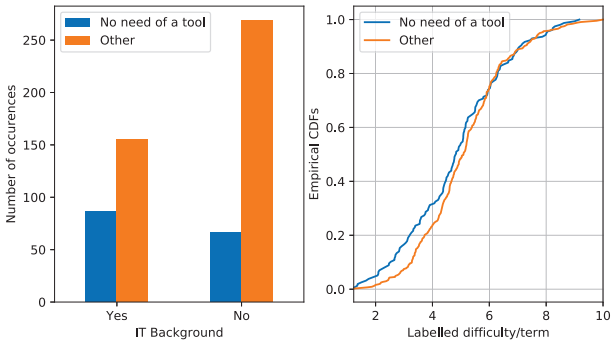


Figure 8: Two factors show significant differences between the participants who did not need a tool to help read security texts and other participants who preferred a tool.

- *"A simple tool that linked to the Wikipedia page or some other article when clicking on terms would be useful. That way if I wanted to learn more, it would be simple to do so."*

We further explored the reasons why 27% of the participants did not need a tool to assist their comprehension. We compared these participants to the rest who preferred an aid tool in terms of their demographics and labelling behaviours. The demographics we collected include gender, age, education, IT background, and whether they are native speakers. Only IT background shows a statistically significant difference between the numbers of people in the two groups ($\chi^2 = 18.563$, $p < 0.001$). Additionally, we calculated the average number of annotations per article and the average difficulty level per annotation for each participant and applied these two measures as labelling behaviours to compare the two groups again. A weak difference is measured with t-test in labelled difficulties ($t = -1.664$, $p = 0.097$). As shown in Fig.8, the users who did not need a tool to help read security texts has a significantly higher proportion with IT background and tend to give fewer annotations and lower difficulty for technical terms, compared to the rest users who would like a tool.

Based on Experiment 1, we conclude:

- The majority of the technical terms are hard for users to understand;
- Traditional readability tests fail to provide consistent difficulty levels with users' reported ones;
- People with IT background give higher difficulty levels for the technical terms related to cyber threats and protection measures than people without IT background;
- Most users would like a dictionary-based aid tool to help read security texts.

4 EXPERIMENT 2: THE EFFECT OF AID TOOLS ON USERS' UNDERSTANDING OF SECURITY TEXTS

4.1 Setup

In this section, we present the generation of a security-centric dictionary (SC Dictionary) as a proof of concept for the framework shown in Fig.1. To test the effectiveness of the dictionary, we developed a service as an add-on tool. From the first experiment, we found that around 65% of the participants would like to have a dictionary to obtain an explanation of the meanings of the terms. Therefore, we developed a security-centric assistant that automatically detects technical terms and shows pop-up descriptions for them. This experiment aims to answer the research questions below regarding the effectiveness of such tools or services in promoting users' understanding.

RQ5. How much does our tool help users understand security texts?
RQ6. What influences users' comprehension of security texts?
RQ7. Does having IT background help understand security texts better?

4.1.1 Making a Security-Centric Dictionary. Some prior efforts have been made to display short descriptions for difficult words (e.g., by Wikipedia Page Previews [59], Google Dictionary [22] and Mac Dictionary [4]). These tools open pop-ups with meanings triggered by text hovering or clicking to help users understand unfamiliar words without the need to open multiple tabs.

By connecting to their provided APIs, we implemented a program in Python to look up the definitions of the corpus terms in their dictionary. Only 27% of the corpus terms sent to Wikipedia Page Previews returned results. Google Dictionary performed slightly better, with a percentage of 35%. The Mac built-in dictionary had less technical knowledge in cybersecurity, and only returned 17% of the definitions. We concluded that the state-of-the-art tools did not perform well on the SC Corpus as they were designed mainly for common words. Therefore, a security-centric dictionary was needed for technical articles.

To build such a dictionary, we implemented a crawler in Python to return the query results of searching term meanings. For ambiguous terms, we also combined them with specific keywords such as 'computing' or 'security' to refine the definitions. Some external websites, during the Google search, also provided supplementary resources. For example, the technology-specific websites, such as Whatis[5], provide technical definitions in IT.

We collected the meanings and image URLs for the terms in the SC Corpus from all available online sources and manually selected the most accurate meaning(s) for each term. For the terms with only textual descriptions, we added a default image too. As a result, we obtained the SC Dictionary, which provided descriptive definitions as well as images for the whole corpus.

The dictionary was leveraged as a knowledge base for our tool. Each term in the dictionary was saved as a JSON file, along with its details (e.g., meaning, image URL, image resolution). We stored the data in an accessible server so that they can be retrieved through 'GET' requests over HTTP.

4.1.2 Making a Security-Centric Assistance Tool. We built our tool on top of SC Dictionary and implemented it as an extension/add-on in the user interface to provide meanings automatically. The extension highlighted the technical terms and used pop-up windows to show the meanings. It was developed with JavaScript and

[5]https://whatis.techtarget.com/

Table 1: Three reading tasks of nine articles and their mentioned security threats.

Task	Article	Discussed Security Issues	#questions in Threats / Protection
1	1	connected car vulnerability, DoS attack, Trojanized apps, bug	11/4
	2	CVE, authentication/VPN/ScreenOS vulnerability, backdoor	
	3	phishing, data breach, ransomware, cryptomining, email-served malware	
2	1	smartwatch/DNS/authentication vulnerability, privacy issues, firmware security	6/9
	2	zero-day vulnerability, chip flaw, CVE	
	3	DoS attack, security log data loss	
3	1	password stealing, phishing, the man-in-the-middle attack	9/6
	2	vulnerability, zero-day threat, domain fraud, indicators of compromise	
	3	Gh0st remote access Trojan, PowerRatankba, WannaCry ransomware attack	

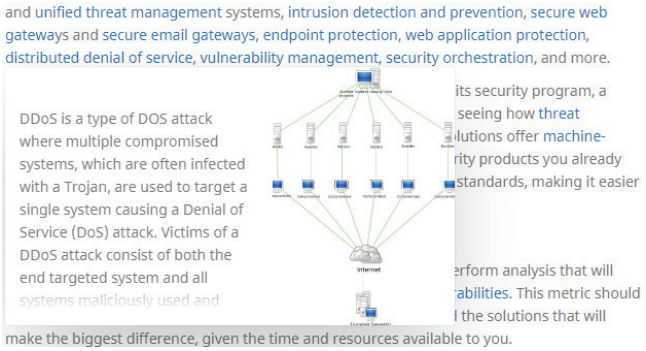


Figure 9: An example of a definition in a pop-up when hovering over the term 'distributed denial of service'.

was compatible with major browsers including Chrome, IE, Firefox and Safari. A screenshot of the tool while describing the term 'distributed denial of service' is shown in Fig.9.

The articles were firstly tokenised and split into sentences by the Stanford Core NLP [33]. We detected the technical terms in the articles by one-to-one mapping after lemmatisation based on the dictionary. Appositions of technical terms detected by dependency extraction were also considered technical. Abbreviations were detected in our extension. For example, 'Two-Factor Authentication' was in our dictionary, but '2FA' was not. However, the tool recognised and treated 2FA similarly.

The detected terms were then highlighted and became clickable by adding the 'a' HTML tags. The properties of the tags were also set to point to the descriptions of the corresponding terms through our designed API connection.

4.2 Evaluation

To evaluate the effectiveness of our tool, we conducted a subjective study of users' understanding of technical articles when using our tool and other methods (e.g., Google Search, pop-up based Google Dictionary [22]).

4.2.1 Questionnaire Design and Implementation. We designed a questionnaire to measure how well the participants understood the technical articles. The evaluation method was adopted from [8]. We selected nine articles from the original dataset, excluding the 200 previously-used articles. The selected articles were distinguishable as they addressed different cybersecurity problems. They were further grouped into three reading tasks randomly. The security threats addressed in these tests are listed in Table 1. Each article was accompanied with five multiple-choice questions about the conceptual understanding of technical terms. Users who can answer our questions correctly are considered to be knowledgeable about security threats and their corresponding solutions. They are considered to be more sensitive and aware of security risks with daily used software applications (e.g., Microsoft Office) or smart devices (e.g., smartwatches). When they face an attack, they are also more likely to purchase security products to protect their authentication or their cloud environment even without additional professional instructions. Our supplementary document[6] explains how users can learn security knowledge to improve their security awareness from each article.

Each task included 15 questions about Security Threats and Security Protection, as shown in Table 1. The two categories are explained as follows:

Security Threats: These questions required participants to comprehend the threats described in the article, such as malicious activity, attack, vulnerability and data breach incident. There were two subcategories for these questions: meaning and function understanding. Meaning understanding required users to select the correct attack from a set of options based on the definition, while the other options provided some similar attacks as wrong answers. The other category tested if participants understood how attacks worked. For example, the user could be asked to select the core technique (algorithm) to exploit a given vulnerability.

Security Protection: These questions referred to defensive solutions against attacks, such as cloud security services or two-factor authentication (2FA). The subcategories were similar to those of Security Threats. Moreover, we listed simulated real-world cases for selection. For instance, we asked the users to select the possible cases of 2FA (e.g., the case of using the password and one-time code sent through SMS); "password only" was among the wrong options.

Our questionnaire[7] included six tests, providing two versions (i.e., plain text and the text with our pop-ups) for each of the three tasks. The participants were assigned the tasks, and the experimental group was provided with the pop-up meanings (test 1-3). The control group was only given plain texts in test 4-6. However, they were free to use any tools or search engines such as Google Dictionary [22] and Google Search. After each submitted the answer sheet, the accuracy and the time spent on the task were automatically calculated and displayed. Each task was designed to be completed by three participants with IT background, and three without IT background. Before launching the experiment, a pilot

[6] https://ktd4869.github.io/Reading_Test_MT/Survey2_explanations.pdf
[7] https://ktd4869.github.io/Reading_Test_MT/

study was also conducted by three people with IT background and another three people without IT background to test the suitability and difficulty of the questionnaire.

4.2.2 Experiment Procedure. We invited the individuals who participated in our IT background test published in MTurk and divided them in to experiment and control groups. Like in the first experiment, users were considered having IT background if they achieved at least an undergraduate degree in IT or 1-year related working experience. We then published our questionnaire in MTurk, with the same amount of assignments released for the link of each test. Each participant was allocated one link randomly, and the completion was rewarded 2 U.S. dollars.

Before running the experiment, we gave a short demo and showed the experimental group (the users who were going to use our tool) how to use our tool. They were asked to answer the questions only with the pop-up meanings generated by our tool. We also showed the control group how to use other methods including a series of search engines (e.g., Google, Wiki) and similar dictionary-based tools (e.g., Wikipedia Page Previews [59], Google Dictionary [22] and Mac Dictionary [4]).

The participants were then required to click the external link to our reading test. Detailed instructions were provided at the starting page. A timer started once they clicked the 'start' button. The time consumed and the accuracy achieved were displayed after the task was completed, and the 'submit' button was clicked.

After completing the questionnaire, the participants were asked to leave feedback from this experience. The experimental group was required to provide suggestions regarding our tool and if they would like to always have it. The control group was asked about the methods they used, whether they were useful or not, and if they would like a tool to help. We used open card sorting [56] again to group the suggestions from the participants. We also classified them into positive and negative comments, so that we could identify the helpful and useless features of our tool for improvement.

We read all the responses and rejected invalid answers, such as empty ones or responses from people who spent less than 10 minutes. The pilot study showed each task cost at least 15 minutes. We read their feedback to inspect if they completed our tasks carefully. In total, we collected 112 valid answers. We further divided the results into IT and non-IT groups, and all the 12 groups had valid responses ($mean = 9.3$, $std = 4.1$).

4.2.3 Experiment Results. Now, we show the results of the user study to answer our three research questions.

RQ5. Efficiency

Based on [8], we measured the accuracy of participants' answers to the multiple-choice questions and the time spent on those. The accuracy was defined as the percentage of correct answers. Each question only had one correct choice. Our questionnaire computed the accuracy and the time spent automatically after submission. A higher accuracy indicates a better understanding of the articles. Also, we assume that shorter answer times (at equal accuracy) imply better comprehension of the contents.

In Fig.10, the violin plots represent the distributions of accuracy and time spent on questionnaire completion for the three reading tasks, where each subplot represents a comparison between the

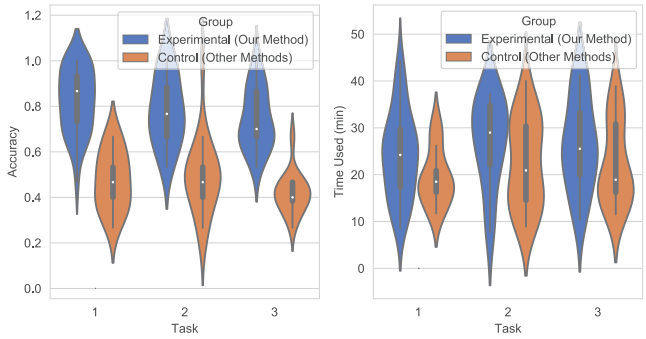


Figure 10: The accuracy and time spent on questionnaire completion for three reading tasks by using our tool or other methods such as Google search.

results by using our tool and other methods (i.e., search engines and other dictionary-based tools). We can see that the accuracy achieved using our tool is significantly higher than that of other methods. With other methods, the control group could only answer around 40% of the questions correctly. This number increases by around 30% in the experimental group for all the three reading tasks. However, the best average accuracy achieved does not go beyond 67%, which means only 10 out of 15 questions are answered correctly. It indicates the difficulty for users to understand security issues comprehensively only by reading technical articles.

The distributions of the time spent on questionnaire completion are demonstrated in the right subplot of Fig.10. We find the average time spent by the experimental group is longer than that of the control group for all the three reading tasks, with 24, 29 and 26 minutes compared to 19, 21 and 19 minutes. Time used by the control group is mainly distributed in two areas with 10 to 20-minute difference of the upper and lower adjacent values. Some users may stop reading and do a web search for definitions. It took more time to open multiple tabs and to find the definitions of technical jargons, especially some ambiguous terms (e.g., host) which have numerous meanings. The smaller amount of time consumed by the control group may result from being impatient and skipping some terms, which could be the keywords in the articles. The three to five-minute difference in time between the two groups is not significant.

We further explored the significance of the difference between the accuracy and the time spent by using our tool as well as other methods. We applied a t-test to measure the difference of accuracy and time spent between the experimental group and the control group. The result shows that the accuracy in each reading task assisted by our tool shows a significant difference to that done by other methods ($p < 0.001$). It implies that our tool can significantly improve users' understanding of the technical articles, while the difference in time spent is not significant between the two groups.

From the feedback of the control group, we find around 61% people in this group did not use any search engine or tool because it would take much longer or they believed they could find the correct answers based on their knowledge. The rest of the group mentioned they used Google or pop-up dictionaries to find the meanings, but most of them only searched for a few terms and felt the searches were not useful for these jargon-laden articles. Only

one participant felt that, while Google was helpful, it would be better to have a tool to provide the definitions by hovering over unfamiliar words. Overall, 80% of participants in the control group preferred to have a tool to support them.

Analysing the comments from the experimental group can help us understand how useful the tool is. We extracted the keywords and grouped the comments to explore their satisfaction regarding the tools. We also reviewed the comments to find what features of our tool were useful to help them understand the articles. From the feedback, we find that all participants in this group deemed the pop-up meanings useful. It helped them understand the articles better and faster. Some participants expressed their satisfaction as:

- *"They were very useful as the definition was in depth."*
- *"I think that pop-up meanings are useful and convey the meaning accurately."*
- *"Yes, the pop-up meanings are VERY useful! They enabled me to see the meanings of words and phrases that I didn't recognise without having to stop reading and do a web search for the definitions. It lets the user actually read the article without stopping to puzzle things out since the pop-ups can be seen as actually being part of the article. They helped tremendously in understanding the article itself.'*
- *"They are very useful especially when you don't understand the keywords or when you get confused."*

There are also some suggestions to help us improve our tool in future work:

- *"I felt like they only should have popped up the first time you saw them. Otherwise, the articles got kind of cramped and wordy. Other than that, they were very useful in explaining exactly what the terms meant."*
- *"They were useful, but I found it difficult to see them entirely. They got cut off at the bottom and I saw no way to scroll."*
- *"They are useful to an extent but the limited windows are annoying when I want more information than what fits in the pop window."*

RQ6. Influential Factors

We further reviewed the questions where our participants got the wrong answers to see what factors influence the accuracy. Potential factors could be the content or question type. The content contained different security issues, including both threats and protection mechanisms. Our questionnaire consisted of both positive and negative questions. We analysed the questions with the wrong answers to find the security issues that are hard to understand. We calculated the frequency of each question wrongly answered in each reading task. If more than a half, we consider it error-prone. We did this calculation for all such questions in the three tasks.

From the results, we find that the number of error-prone questions answered by the control group using other methods is around twice as many as those in the experimental group. We hypothesised that, with the help of our tool, some questions could be easier to answer, especially the conceptual questions for both threats and protection categories. These two include the definitions of specific attacks (e.g., the core algorithms, involved platforms) and protection-related subjects (e.g., relevant techniques/services, development team, practical use cases). However, we find that several questions are error-prone even with the help of our tool. For instance, some questions require the user to understand the meaning

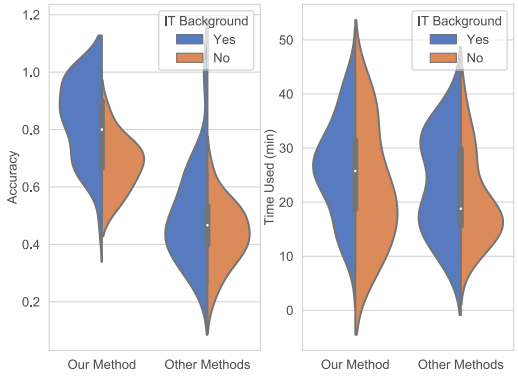


Figure 11: The accuracy and time spent on questionnaire completion of participants with different IT backgrounds.

of an attack under a condition. People need preliminary knowledge of the field to understand such technical articles to some extent. Although each detected term was provided with meanings, participants were confused with multiple similar terms.

RQ7. IT Knowledge Effect

We compared the difference in the accuracy and time spent on questionnaire completion between participants with different IT backgrounds. A Mann-Whitney U test was used to see if the difference is significant between the results of people with different IT backgrounds. We also made a separate comparison between the experimental group and the control group.

Fig.11 presents the violin plots of the accuracy and the time spent between participants with different IT backgrounds. For people with IT background, the average accuracy is around 90% when they use our tool, which is 20% higher than people without IT background. The accuracy drops to 45% when people with IT background use other methods, and the accuracy is similar to people without IT background. We also find that people with IT background read technical articles more slowly than people without IT background, an average of 27 minutes compared with 20 minutes (by using our tool), and 17/32 minutes compared with 17 minutes (by using other methods). From their comments, we find that people with IT background read the security articles more carefully and spent more time reading the explanations in the provided pop-ups compared to people without IT background.

The significance test shows that only the accuracy achieved with our tool between people with different IT backgrounds shows a statistically large difference ($p < 0.001$). It indicates that users with IT background face as much difficulty as normal users do in comprehending security issues. Still, people with IT background understand technical articles significantly better than people without IT background when both of them use our tool.

Based on Experiment 2, we conclude that:

- Our tool can help users understand security texts better, with 30% improvement;
- Users have difficulty in understanding ambiguous terms;
- Users with IT background show significantly better performance in understanding security texts only when they use our tool.

5 DISCUSSION

5.1 Implications

For Researchers. Our user studies were based on the texts extracted from security blogs. Blogs report the latest security trends and advancements, including news, hacks, discoveries, vulnerabilities and their solutions. If users sufficiently comprehend security articles, they are more likely to take reasonable actions to minimise risks when they face a threat. Future research may also explore other online and user-friendly public resources such as videos to raise users' security awareness and understanding. Our findings can also inspire researchers to complement this line of work from alternative viewpoints, for example, difficulty level measures for technical terms, automatic self-explanations for security articles, or replacing technical terms with commonly used explanatory phrases.

For IT Practitioners. Our findings suggest users with IT background do not have much more cybersecurity knowledge than users without IT background. This is consistent with other research which showed that they are likely to share confidential forms or download unreliable software without consulting security specialists [24]. Intermedia's Insider Risk Report [28] revealed that tech-savvy workers are more likely to create security risks. IBM's 2016 Cyber Security Intelligence Index also found that of all cyber-attacks reported, 60% of them were caused by insiders, among which 25% were the result of employee negligence [47]. It also reported that IT workers usually overestimate their ability to defend against attacks. Security awareness programs for IT employees are also considered as highly important. Our tool can assist in enhancing users' awareness and understanding.

For Educators. Our survey results revealed that users have only limited security knowledge to protect themselves from attacks. Educating users can help them perform better in risk perception and understanding. Blogs are widely accessible and provide end users with timely information. Our tool was designed with educational purposes in mind, to help users increase their security knowledge. With friendly integration to browsers, a convenient reading assistant promotes users' interest in security news and articles. Additionally, security descriptions are usually too technical and difficult for home users such as security advice and privacy policy statement. Our findings suggest that, without appropriate explanation, users tend to skip the keywords, which might be the crucial hints. With aid tools, such as our pop-up dictionary, users can quickly get the point of security advice and follow its instructions.

5.2 Limitations

User study. Our SC Corpus was generated by human annotations from 200 representative articles with an average length of 1,000 words each. Due to our limited budget, we only recruited 597 crowdworkers to annotate the terms. Future work can employ more articles and annotators to build a larger corpus. The sample set in our study might not have the same population diversity as found in more massive sample sets. But our analysis revealed the significant difference between randomly divided groups. This study is instructive for future work to involve a larger number of participants.

Tool Development. The tool we developed is simple but proven to be effective. However, there is still room for improvement. Firstly, our pop-up windows were limited in size. A few terms have relatively long descriptions, and they were cut at the end. Future work can extract the most informative words or sentences to shorten the descriptions. Secondly, our tool might not do well for newly created terms since it was designed based on the SC corpus from the user study. As we used more than 40 thousand technical articles from 2014 to the present, our tool can still be used in the majority of current security articles. Future work can detect new terms based on similarity to our knowledge base. The last limitation is in the evaluation. With using our tool, the terms highlighted in the articles could serve bias users, as it draws more attention to these terms compared to the rest text. As there is no real incentive for the participants to answer correctly, the results might not reflect their best efforts. Future research should consider real-world practice.

6 CONCLUSION

In this paper, we studied users' understanding of security and how well they comprehend the security related articles. We found that most participants had difficulty understanding the technical terms of the articles related to security. Based on a crowdsourcing task, we generated a security-centric corpus with more than five thousand terms. We also developed a tool to help users understand security articles by displaying meanings for technical terms in pop-ups. An experiment demonstrated the pop-up explanations greatly improved users' security understanding. Our analysis also revealed users with IT background did not understand security articles better or faster than people without IT background. Users' misconceptions of cybersecurity issues may hinder security controls application or lead to misuse of security measures.

Inspired by our findings, we proposed several future research directions. A larger number of crowdworkers can be employed to annotate more security articles to generate a broader and richer security-centric corpus. Future research should attempt to create refined meanings for the terms. Since end users have different levels of education, one solution is to provide them with personalised explanations. Instead of plain texts, visual aids such as infographics can also be studied to explain security knowledge in a user-friendly way.

REFERENCES

[1] Yasemin Acar, Sascha Fahl, and Michelle L Mazurek. 2016. You are not your developer, either: A research agenda for usable security and privacy research beyond end users. In *SecDev'16*. IEEE, 3–8.

[2] Alessandro Acquisti, Idris Adjerid, Rebecca Balebako, Laura Brandimarte, Lorrie Faith Cranor, Saranga Komanduri, Pedro Giovanni Leon, Norman Sadeh, Florian Schaub, Manya Sleeper, et al. 2017. Nudges for privacy and security: understanding and assisting users' choices online. *Comput. Surveys* 50, 3 (2017), 44.

[3] Bonnie Brinton Anderson, C Brock Kirwan, Jeffrey L Jenkins, David Eargle, Seth Howard, and Anthony Vance. 2015. How polymorphic warnings reduce habituation in the brain: Insights from an fMRI study. In *CHI'15*. ACM, 2883–2892.

[4] Apple. cited Nov 2019. Mac Dictionary. https://support.apple.com/en-au/guide/dictionary/dictionary-user-guide-dic34880/mac.

[5] Nalin Asanka Gamagedara Arachchilage and Steve Love. 2013. A game design framework for avoiding phishing attacks. *Computers in Human Behavior* 29, 3 (2013), 706–714.

[6] Nalin Asanka Gamagedara Arachchilage, Steve Love, and Konstantin Beznosov. 2016. Phishing threat avoidance behaviour: An empirical investigation. *Computers in Human Behavior* 60 (2016), 185–197.

[7] Maria Bada, Angela M Sasse, and Jason RC Nurse. 2019. Cyber security awareness campaigns: Why do they fail to change behaviour? *arXiv preprint arXiv:1901.02672* (2019).

[8] Lingfeng Bao, Zhenchang Xing, Xin Xia, and David Lo. 2018. VT-Revolution: Interactive Programming Video Tutorial Authoring and Watching System. *IEEE Transactions on Software Engineering* (2018).

[9] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.

[10] Pew Research Center. cited Nov 2019. What Internet Users Know about Technology and the Web. https://www.pewinternet.org/2014/11/25/web-iq/.

[11] Netwrix Corporation. cited Nov 2019. 2017 IT Risks Report. https://www.netwrix.com/2017itrisksreport.html.

[12] Damien Cram and Béatrice Daille. 2016. Terminology extraction with term variant detection. In *Proceedings of ACL-2016 System Demonstrations*. 13–18.

[13] Lorrie F Cranor. 2008. A framework for reasoning about the human in the loop. (2008).

[14] David M Diez, Christopher D Barr, and Mine Cetinkaya-Rundel. 2012. *OpenIntro statistics*. OpenIntro.

[15] W Keith Edwards, Erika Shehan Poole, and Jennifer Stoll. 2008. Security automation considered harmful?. In *Proceedings of the 2007 Workshop on New Security Paradigms*. ACM, 33–42.

[16] Lisa Maszkiewicz Rock Stevens Everest Liu Dhruv Kuchhal Elissa M. Redmiles, Miraida Morales and Michelle L. Mazurek. 2015. First Steps Toward Measuring the Readability of Security Advice. In *Workshop on Technology and Consumer Protection*.

[17] Michael Fagan and Mohammad Maifi Hasan Khan. 2016. Why do they do what they do?: A study of what motivates users to (not) follow computer security advice. In *SOUPS'16*. 59–75.

[18] Adrienne Porter Felt, Alex Ainslie, Robert W Reeder, Sunny Consolvo, Somas Thyagaraja, Alan Bettes, Helen Harris, and Jeff Grimes. 2015. Improving SSL warnings: Comprehension and adherence. In *CHI'15*. ACM, 2893–2902.

[19] Darja Fišer, Vit Suchomel, and Miloš Jakubícek. 2016. Terminology Extraction for Academic Slovene Using Sketch Engine. *RASLAN 2016 Recent Advances in Slavonic Natural Language Processing* (2016), 135.

[20] Dinei Florêncio, Cormac Herley, and Paul C Van Oorschot. 2014. An administrator's guide to internet password research. In *LISA'14*. 44–61.

[21] Alain Forget, Sarah Pearman, Jeremy Thomas, Alessandro Acquisti, Nicolas Christin, Lorrie Faith Cranor, Serge Egelman, Marian Harbach, and Rahul Telang. 2016. Do or do not, there is no try: user engagement may not improve security outcomes. In *SOUPS'16*. 97–111.

[22] Google. cited Nov 2019. Google Dictionary. https://chrome.google.com/webstore/detail/google-dictionary-by-goog/mgijmajocgfcbeboacabfgobmjgjcoja?hl=en.

[23] Brij Gupta, Dharma P Agrawal, and Shingo Yamaguchi. 2016. *Handbook of research on modern cryptographic solutions for computer and cyber security*. IGI Global.

[24] David R Hannah and Kirsten Robertson. 2015. Why and how do employees break and bend confidential information protection rules? *Journal of Management Studies* 52, 3 (2015), 381–413.

[25] Marian Harbach, Markus Hettig, Susanne Weber, and Matthew Smith. 2014. Using personal examples to improve risk communication for security & privacy decisions. In *CHI'14*. ACM, 2647–2656.

[26] Anna Hätty, Simon Tannert, and Ulrich Heid. 2017. Creating a gold standard corpus for terminological annotation from online forum data. In *LOTKS'17*.

[27] Adele E Howe, Indrajit Ray, Mark Roberts, Malgorzata Urbanska, and Zinta Byrne. 2012. The psychology of security for the home computer user. In *2012 IEEE Symposium on Security and Privacy*. IEEE, 209–223.

[28] Intermedia. cited Nov 2019. Intermedia's 2015 Insider Risk Report. https://www.intermedia.net/report/riskiestusers.

[29] Iulia Ion, Rob Reeder, and Sunny Consolvo. 2015. "... No one Can Hack My Mind": Comparing Expert and Non-Expert Security Practices.. In *SOUPS'15*, Vol. 15. 1–20.

[30] Iacovos Kirlappos and M Angela Sasse. 2011. Security education against phishing: A modest proposal for a major rethink. *IEEE Security & Privacy* 10, 2 (2011), 24–32.

[31] Katharina Krombholz, Wilfried Mayer, Martin Schmiedecker, and Edgar Weippl. 2017. " I Have No Idea What I'm Doing"-On the Usability of Deploying {HTTPS}. In *USENIX Security'17*. 1339–1356.

[32] Xiaojing Liao, Kan Yuan, XiaoFeng Wang, Zhou Li, Luyi Xing, and Raheem Beyah. 2016. Acing the IOC game: Toward automatic discovery and analysis of open-source cyber threat intelligence. In *CCS'16*. ACM, 755–766.

[33] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*. 55–60.

[34] Stephen Mayhew and Dan Roth. 2018. Talen: Tool for annotation of low-resource entities. In *Proceedings of ACL 2018, System Demonstrations*. 80–86.

[35] Susan E McGregor, Polina Charters, Tobin Holliday, and Franziska Roesner. 2015. Investigating the computer security practices and needs of journalists. In *USENIX Security'15*. 399–414.

[36] Nikki McNeil, Robert A Bridges, Michael D Iannacone, Bogdan Czejdo, Nicolas Perez, and John R Goodall. 2013. Pace: Pattern accurate computationally efficient bootstrapping for timely discovery of cyber-security concepts. In *ICMLA'13*, Vol. 2. IEEE, 60–65.

[37] Medium. cited Nov 2019. Read Time and You. https://blog.medium.com/read-time-and-you-bc2048ab620c.

[38] Alena Naiakshina, Anastasia Danilova, Christian Tiefenau, Marco Herzog, Sergej Dechand, and Matthew Smith. 2017. Why do developers get password storage wrong?: A qualitative usability study. In *CCS'17*. ACM, 311–328.

[39] Anand Narasimhamurthy. 2005. Theoretical bounds of majority voting performance for a binary classification problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27, 12 (2005), 1988–1995.

[40] Eyal Peer, Joachim Vosgerau, and Alessandro Acquisti. 2014. Reputation as a sufficient condition for data quality on Amazon Mechanical Turk. *Behavior research methods* 46, 4 (2014), 1023–1031.

[41] Carol Peters. 2005. *Multilingual Information Access for Text, Speech and Images: 5th Workshop of the Cross-Language Evaluation Forum, CLEF 2004, Bath, UK, September 15-17, 2004, Revised Selected Papers*. Vol. 3491. Springer Science & Business Media.

[42] Emilee Rader and Rick Wash. 2015. Identifying patterns in informal sources of security information. *Journal of Cybersecurity* 1, 1 (2015), 121–144.

[43] Paul Rayson and Roger Garside. 2000. Comparing corpora using frequency profiling. In *Proceedings of the workshop on Comparing corpora-Volume 9*. Association for Computational Linguistics, 1–6.

[44] Elissa M Redmiles, Sean Kross, and Michelle L Mazurek. 2016. How i learned to be secure: a census-representative survey of security advice sources and behavior. In *CCS'16*. ACM, 666–677.

[45] Elissa M Redmiles, Amelia R Malone, and Michelle L Mazurek. 2016. I think they're trying to tell me something: Advice sources and selection for digital security. In *Proceedings of S&P'16*. IEEE, 272–288.

[46] Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. ELRA, Valletta, Malta, 45–50. http://is.muni.cz/publication/884893/en.

[47] IBM X-Force® Research. cited Nov 2019. Reviewing a year of serious data breaches, major attacks and new vulnerabilities. https://www.triscal.com.br/wp-content/uploads/2018/08/file_seguranca-ibm-xforce-cyber_index-2016.pdf.

[48] Leonard Richardson. cited Nov 2019. Beautiful Soup. https://www.crummy.com/software/BeautifulSoup.

[49] Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the space of topic coherence measures. In *WSDM'15*. ACM, 399–408.

[50] Dale C Rowe, Barry M Lunt, and Joseph J Ekstrom. 2011. The role of cyber-security in information technology education. In *SIGITE'11*. ACM, 113–122.

[51] Scott Ruoti, Tyler Monson, Justin Wu, Daniel Zappala, and Kent Seamons. 2017. Weighing context and trade-offs: How suburban adults selected their online security posture. In *SOUPS'17*. 211–228.

[52] Dominik Sacha, Hansi Senaratne, Bum Chul Kwon, Geoffrey Ellis, and Daniel A Keim. 2016. The role of uncertainty, awareness, and trust in visual analytics. *IEEE transactions on visualization and computer graphics* 22, 1 (2016), 240–249.

[53] Yukiko Sawaya, Mahmood Sharif, Nicolas Christin, Ayumu Kubota, Akihiro Nakarai, and Akira Yamada. 2017. Self-confidence trumps knowledge: A cross-cultural study of security behavior. In *CHI'17*. ACM, 2202–2214.

[54] Florian Schaub, Rebecca Balebako, Adam L Durity, and Lorrie Faith Cranor. 2015. A design space for effective privacy notices. In *SOUPS'15*. 1–17.

[55] Richard Shay, Iulia Ion, Robert W Reeder, and Sunny Consolvo. 2014. My religious aunt asked why i was trying to sell her viagra: experiences with account hijacking. In *CHI'14*. ACM, 2657–2666.

[56] Donna Spencer. 2009. *Card sorting: Designing usable categories*. Rosenfeld Media.

[57] Blase Ur, Jonathan Bees, Sean M Segreti, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2016. Do Users' Perceptions of Password Security Match Reality?. In *CHI'16*. ACM, 3748–3760.

[58] Blase Ur, Fumiko Noma, Jonathan Bees, Sean M Segreti, Richard Shay, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2015. " I Added'!'at the End to Make It Secure": Observing Password Creation in the Lab. In *SOUPS'15*. 123–140.

[59] Olga Vasileva. cited Nov 2019. Wikipedia Page Previews. https://blog.wikimedia.org/2018/05/09/page-previews-documentation/.

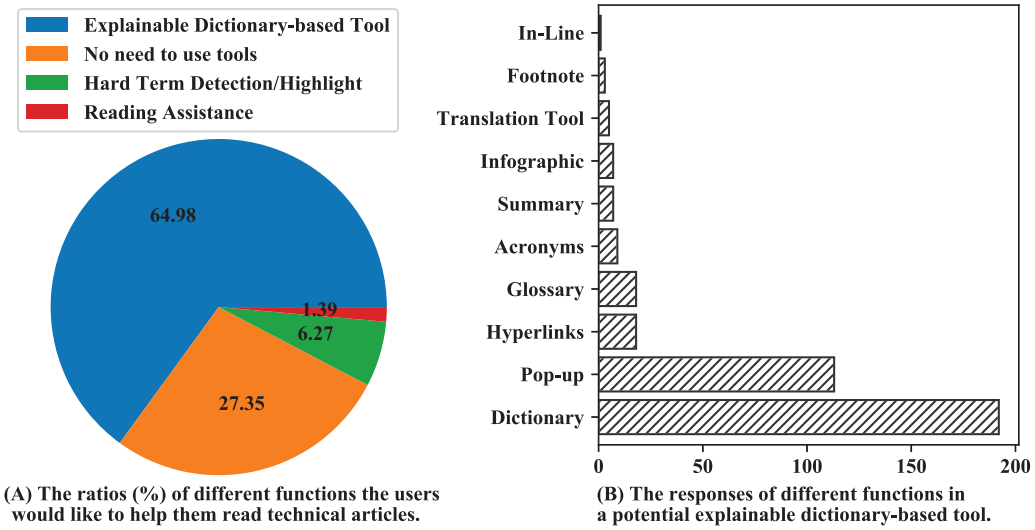
[60] Rick Wash. 2010. Folk models of home computer security. In *SOUPS'10*. ACM, 11.

[61] Rick Wash and Emilee J Rader. 2015. Too Much Knowledge? Security Beliefs and Protective Behaviors Among United States Internet Users.. In *SOUPS'15*. 309–325.

[62] Le Yu, Tao Zhang, Xiapu Luo, Lei Xue, and Henry Chang. 2017. Toward automatically generating privacy policy for android apps. *IEEE Transactions on Information Forensics and Security* 12, 4 (2017), 865–880.

[63] Mu Zhang, Yue Duan, Qian Feng, and Heng Yin. 2015. Towards automatic generation of security-centric descriptions for android apps. In *CCS'15*. ACM, 518–529.

[64] Yixin Zou, Abraham H Mhaidli, Austin McCall, and Florian Schaub. 2018. I've got nothing to lose: consumers' risk perceptions and protective actions after the equifax data breach. In *SOUPS'18*. 197–216.



(A) The ratios (%) of different functions the users would like to help them read technical articles.

(B) The responses of different functions in a potential explainable dictionary-based tool.

Figure 12: The functions users would like to help read technical articles from our survey.

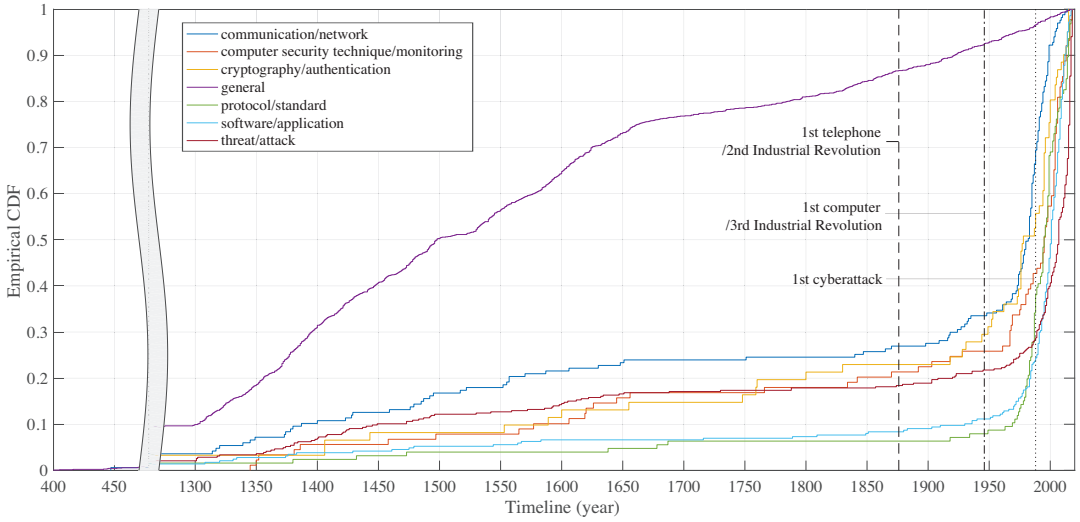


Figure 13: Empirical CDFs of the terms in 7 of our categories.

A FIGURES.

A.1 History analysis of technical terms.

We compare the empirical CDFs of the terms in 7 of our categories. Since most of the terms in the general category are common and ordinary, we use the CDF of the general category as the baseline. In Fig.13, we show that the terms were first coined after the year 400, and the number of the terms increase slowly until the year 1300. The empirical CDFs in computer security technique/monitoring, cryptography/authentication, and threat/attack are close to each other, which indicates a similar history between the terms related to 'security' and 'attack'. Besides, the figure shows that over 85% terms in software/application and about 90% terms in protocol/standard appeared after the invention of the first computer in 1946. Similarly, there are rapid increases after 1946 for computer security technique/monitoring, cryptography/authentication, and threat/attack. Since the first cyberattack happened in the 1980s, computer security/attacks have received very high attention with exponential growth in the types and number.

Except for the general category, the empirical CDF of the communication/network is higher than the others. About 30% of the terms in the communication/network category emerged before the nineteenth century. That is because communication is an old concept, and some terms (e.g., transmit, session, and route) have been used since the middle ages. After a two-hundred-year stable phase, the number of the terms in communication/network suddenly increases following the first invention of the telephone, which is also the beginning of the Second Industrial Revolution.

B TABLE.

Table 2: 15 categories of the security terms and their detailed descriptions.

	Categories	Sub-categories	Examples	Description
1	algorithm/novel technology (in business)	algorithm	MD5, Network segmentation	Computer algorithm, used to perform calculation, data processing, automated reasoning, and other tasks.
		machine intelligence	AI, machine learning	Intelligence demonstrated by machines, in contrast to the natural intelligence displayed by humans and animals.
		novel technology (in business)	E-Business, smart home	Recent proposed techniques, might be used in business
2	computer security technique/monitoring	computer security technique	Check Malicious utility from Kutools, DDoS protection	The protection of computer systems from theft or damage to their hardware, software or electronic data, as well as from disruption or misdirection of the services they provide.
		monitoring	alert, monitors online activity	Network or system monitoring.Collecting and analysing information to detect suspicious behavior, unauthorised system changes on the network or events that occur in an operating system or other software runs.
3	cryptography /authentication	authentication method	2FA, CAPTCHA	Verifying the identity of someone (a user, device, or an entity) who wants to access data, resources, or applications
		cryptography	128-bit keys, BoringSSL	Techniques for secure communication in the presence of third parties called adversaries.
4	data structure/file format	data structure	32bit integers, ASCII character	Specialized format for organizing and storing data.
		file format	APK file, DAT file	Standard way that information is encoded for storage in a computer file.
5	general	general	3D, address	Words or phrases used in general purpose.
		number	0x0FFFFFFF	Quantity or amount.
6	hardware	electronic systems and computing	Atmel ATMEGA8, CPU	Components that control a device, specifically configuration, installation and repair control systems, such as avionics, telephone systems and computer systems.
		hardware	Bluetooth devices, CCTV	Collection of physical parts of a computer system.
7	others	ambiguous	agency, agreement	Terms with multiple meanings in computer science. Context is needed to identify.
		computer measure	1024MB, 3 log2N qubits	Computer storage and memory.
		cryptocurrency	Bitcoin, BTC	Digital asset designed to work as a medium of exchange that uses strong cryptography to secure financial transactions, control the creation of additional units, and verify the transfer of assets.
		mobile	Android, iPhone	Mobile phones, handheld computers, and similar technology.
		permission (file/user)	access, administrator privileges	The authorization given to users that enables them to access specific resources on the network, such as data files, applications, printers and scanners.
		radiofrequency	860MHz, Doppler effect	A frequency or band of frequencies in the range 104 to 1011 or 1012 Hz, suitable for use in telecommunications.
		undefined	AdminMailVendorI, Alejandro	Hard to find specific decription.
8	programming /command /operation	command/operation	AND operation, Config	Specific instruction given to a computer application to perform some kind of task or function. Data processing in which the result is completely specified by a rule.
		programming	AJAX, HTML	Process of designing and building an executable computer program for accomplishing a specific computing task.
9	software/application	platform/service	Amazons EC2, ColdFusion	Platform or service provided for customers that supports the development, running, and management of applications.
		software/application	7Zip, Apple Pay	The programs and other operating information used by a computer.
10	protocol/standard	protocol	BLE, HTTPS	General or communication protocol. A set of rules or procedures for transmitting data between electronic devices, such as computers, or the original draft of a diplomatic document, especially of the terms of a treaty agreed to in conference and signed by the parties.
		standard (encryption, character encoding, act, security)	AES, base64	Standard or specification for encryption, character encoding, act, or security.
11	threat/attack	concealment	backdoor, BlackEnergy	Malware designed to operate undetected, not sabotage and ransomware.
		cyberwarfare	cyber espionage, Guccifer 2.0 persona	The use or targeting in a battlespace or warfare context of computers, online control systems and networks.
		data breach	Equifax breach, privilege escalation	The intentional or unintentional release of secure or private/confidential information to an untrusted environment.
		hacker/threat actor	APT28, Ardit	Malicious actor is a person or entity that is responsible for an event or incident that impacts, or has the potential to impact, the safety or security of another entity.
		hacking tool (Vulnerability, Forensics OS, Exploit Payload Social engineering)	0day, CVE20150984	The "system" under attack may be anything from a single application, through a complete computer and operating system, to a large network.
		infectious malware	Disk Wiper Malware, mail worm	Stand-alone malware software that actively transmits itself over a network to infect other computers.
		malware for profit	botnet, CrypMIC	Programs designed to monitor users' web browsing, display unsolicited advertisements, or redirect affiliate marketing revenues to the spyware creator.
		personal data/credential information	credit card data, online banking credentials	Information that relates to an identified or identifiable living individual
		threat technique	Angler exploit kit, blind SQL injections	Techniques or algorithms used to perform attacks.
		other threat/attack	active jamming using an RFCat, compromise Oracle HTTP Server	Other unidentified attacks.
12	communication /network	communication/network	adhoc, CDMA	The transmission of this digital data between two or more computers and a computer network or data network is a telecommunications network that allows computers to exchange data.
13	company /organisation /website /conference /team	company/organisation/website /conference /team	Acer, Alienware	Proper noun (company, organisation, website, conference, or team). Mostly IT related.
14	computer role	computer role	ADMIN, CISO (Chief information security officer)	A group of computers in a zone with a set of role assignments to users or groups
15	software development	software development	deployment pipelines, system integration	The process of conceiving, specifying, designing, programming, documenting, testing, and bug fixing involved in creating and maintaining applications, frameworks, or other software components.