"I want my money back!" Limiting Online Password-Guessing Financially

Maximilian Golla
Horst Görtz Institute
Ruhr-University Bochum
Bochum, Germany
maximilian.golla@rub.de

Daniel V. Bailey
Horst Görtz Institute
Ruhr-University Bochum
Bochum, Germany
danbailey@sth.rub.de

Markus Dürmuth
Horst Görtz Institute
Ruhr-University Bochum
Bochum, Germany
markus.duermuth@rub.de

ABSTRACT

Online password guessing attacks are a serious threat to the integrity of online accounts. A common defense is rate-limiting, either by slowing down or blocking connections, or by requiring CAPTCHAs to be solved. Either of these options has serious drawbacks, facilitating denial of service attacks, being circumventable by proxies and CAPTCHA solving services, and offering bad usability to the legitimate user. Furthermore, guessing attacks are becoming increasingly easier, fueled by recent data breaches containing several hundred million credentials from famous websites.

In this work-in-progress report, we propose an opt-in deposit-based approach to rate-limiting that tackles online guessing attacks. By demanding a small deposit for each login attempt, which is immediately refunded after a successful sign in, online guessing attackers face high costs for repeated unsuccessful logins. We provide an initial analysis of suitable payment systems and reasonable deposit values for real-world implementations and discuss security and usability implications of the system.

1. INTRODUCTION

User authentication is an essential requirement for modern websites as more and more access-controlled services move online. Passwords are widely used for user authentication in today's web services, but human-chosen passwords are far from being secure. Even worse, recently data breaches containing several hundred million credentials from famous websites like Yahoo, MySpace, LinkedIn, Twitter, Dropbox, and VK became public [16]. Consequently, traditional password guessing accelerated by password reuse checking is used by attackers for account takeover [25]. In a password-guessing attack an adversary (i) guesses a password, (ii) verifies its correctness, and (iii) repeats the steps until the correct password is found. Usually, attackers can only try a limited number of guesses from a single IP address to avoid rate-limiting. NIST [5, 20] proposes to limit the number of allowed login attempts to 100 within a period of 30 days. Using proxy servers, the Tor network, or botnets allows the attacker to circumvent these rate-limiting mechanisms. Especially, CAPTCHA-based rate-limiting [39] is frequently used, but can often be neutralized by the use of automatic solving services [31].

We propose a deposit-based rate-limiting approach that requires a deposit before one is able to login. After paying, the actual authentication procedure takes place, i. e., by entering a password. Honest users can expect an immediate refund after the login succeeded. In contrast, malicious attempts by guessing online attackers are not refunded. We follow an opt-in approach that allows a partial roll-out. Such systems rely on the availability of common, instant, and cheap micropayment technologies. Recent developments [8, 28, 32] of blockchain-based payment systems that offer the ability to close smart contracts [38] and their browser integration [12] enable real-world implementations.

Specifically, our contributions include: (i) A novel rate-limiting approach based on paying a deposit before being allowed to log in. (ii) A description of the preconditions for a payment system and initial discussion of parameter choice. (iii) An initial discussion of the security and usability implications and the overall practicability of such a system.

2. RELATED WORK

Closest to our proposal is a system referred to as "payment(s) at risk" proposed by Abadi et al. [1] in 2003 that tries to prevent unsolicited emails (spamming). Its main idea has also been suggested for spam prevention in VoIP networks (SPIT) [10, 33]. In contrast to those proposals, we follow an opt-in approach that allows a partial roll-out.

Alsaleh et al. [2] analyzed login protocols designed to hinder online guessing attacks. In their work, they describe a new login protocol that uses CAPTCHAs and evaluates the requesting source IP and the existence of cookies, which is more restrictive against online guessing attacks, while safely allowing a large number of failed attempts for legitimate users. Freeman et al. [17] studied a more evolved account takeover protection mechanisms, which evaluates a broad range of cues from traffic, browser, and usage fingerprinting to measure user authenticity. In the case of a suspicious login attempt, they re-enforce the login by challenging an additional security question.

Schechter et al. [36] tries to limit the number of weak accounts but can not prevent the consequences of password reuse by deploying a service-specific password composition policy that rejects too popular passwords.

Symposium on Usable Privacy and Security (SOUPS) 2017, July 12–14, 2017, Santa Clara, California.

Herley and Florêncio [23] suggested creating a large number of fake credentials that will lead to honeypot sessions, which are derived from real user data with fake identification information. Zhao and Mannan [42] proposed giving access to a bogus session containing non-valid user data, if an incorrect password was used. This way legitimate users can be relieved from solving CAPTCHAs, while attackers need to learn how to tell fake and real sessions apart.

Online guessing attackers were studied in the context of personal knowledge questions [4, 35, 34, 21] and password guessing [2, 3]. The security threat of *targeted* online guessing attacks was analyzed by Wang et al. [41]. They found current security mechanisms to be ineffective against such attacks. Some proactive users adopt password managers [7, 29] that can generate long random passwords with no reuse. Furthermore, the introduction of low-effort two-factor authentication [26, 9] can help to enhance account security.

3. DEPOSIT-BASED RATE-LIMITING

In the next section, we describe the adversary model and the basic construction of the rate-limiting idea.

3.1 Adversary Model

We consider an online *trawling* attacker [4], i. e., attacks where an adversary is interested in the takeover of *any* account. (In contrast, in a *targeted* attack the adversary is focused on a single *specific* target [41]). We assume the attacker is financially limited in some form: Either one is trying to profit from the attack by selling data on an underground market, or one is hacking "for fun" but with a limited budget. Stealing credentials is only a first step in the error-prone process of abusing accounts for financial gain. Spending too much money becomes unappealing when accounts are on sale at 5 percent of their actual monetary value [13]. Specifically, our countermeasure can be calibrated to make the account takeover-and-resale business uneconomic, as it renders this class of attack more expensive. We discuss the focus on the adversary in the following.

Websites have deployed countermeasures to discourage naïve online password guessing. These include limiting the number of incorrect password entries before locking an account; and increasing the number of guesses required with password strength meters and password composition policies to encourage users to choose more secure passwords. Trawling attackers are well known in the context of, e. g., answering personal knowledge questions [4, 35, 34, 21] or guessing passwords [2, 3] by giving, for example, the most likely answer based on population-wide statistics. In an online password guessing scenario an attacker is able to exploit the password reuse problem by trying out leaked credentials of website A on another website B. This way, trawling attackers in combination with password reuse pose a more serious threat to the security of the users, than just guessing the most common passwords, which are often blacklisted [14, 22].

Our proposed countermeasure is independent of specific guessing techniques. Whether the attacker guesses common passwords or abuses a password leak to make more specific guesses, our countermeasure tackles what all trawling online guessing attacks have in common: they aim to cover a large number of accounts from many different users. An attacker might still succeed against some accounts, but the overall budget required per account will be higher.

3.2 Description

Our basic idea is to require a deposit before the website accepts a login request. Honest users (entering the correct password) can expect a refund of the deposit after the login succeeds. In contrast, malicious guessing attempts (that test the wrong password) are not refunded. This way, honest users will not actually pay money, whereas trawling online guessing attackers must face high costs for repeated login attempts.

3.2.1 Enrollment

There are no adaptations of existing account registration systems required. Instead, we envision it as an opt-in additional layer of security to one's account similar to mechanisms like "2-Step Verification" [18]. The website may also offer a relaxation of other security mechanisms (such as CAPTCHA-based rate-limiting or risk-scores) if a user opts-in. The user will need a micropayment account and suitable browser extension. These are presently offered in the Brave browser, for example. When enabled, the site operator will ask for a deposit, whenever someone tries to sign into the account from an unfamiliar computer.

3.2.2 Authentication

The authentication phase is extended by one additional step as follows. The complete process is visualized in Figure 1. After providing a username, we ask for a deposit before the user is able to proceed to the password entry form. i) In the case, the payment is not authorized by the user or not received by the website the authentication process does not start. ii) In the case the deposit payment is authorized by the user and received by the website, the user is allowed to authenticate by, e. g., entering a password. A successful authentication leads to a refund of the deposit made; unsuccessful attempts can be repeated at the cost of another deposit. Recent work by Chatterjee et al. [6] shows how to protect users against typos and careless mistakes and prevents unforeseeable debts.

We discuss the requirements for a suitable payment system in Section 4.1 and provide an estimation of reasonable deposit amounts in Section 4.2.

3.2.3 Fallback Authentication

In cases where password recovery by, e. g., out-of-band communication like email, is required to reset the primary authenticator, we follow the standard behavior. Thus, a user is allowed to reset the password without any charge or requirement to pay a deposit. This way, a forgotten password will not result in any disadvantage for the user. We discuss possible implications of refunding accumulated deposits (in the case of past typos) in Section 4.

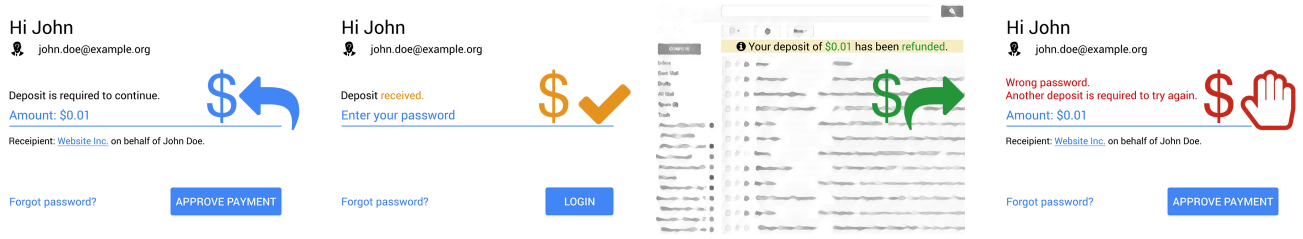
4. DISCUSSION

Next, we discuss options for payment systems and pricing schemes for the proposed rate-limiting approach, as well as benefits and challenges for security and usability.

4.1 Payment System

In the following, we discuss requirements a payment system should fulfill to be suitable for this deposit-based rate-limiting approach.

Viewed as a financial instrument, the fundamental concept of the proposed rate-limiting approach is related to ideas



(a) Step 1: Deposit (b) Step 2: Authentication (c) Step 3: Refund (d) Incorrect: No refund

Figure 1: After providing a username, one is requested to pay a deposit. Once the deposit has been received, one is able to authenticate (i. e., by entering a password). After successful authentication, the deposit is refunded. If the authentication is unsuccessful (i. e., by entering a wrong password), the deposit is not refunded, and every additional attempt will require the payment of another deposit.

from financial engineering [24]. In this system, the user pays a deposit in exchange for the right to submit a password for some fixed interval of time. Here, the funds are held in escrow for the handful of seconds it takes to provide a primary authenticator (i. e., typing a password) and then quickly returned or forfeited as the credential is accepted, rejected, or the interval of time expires. Observe that well-known financial engineering concepts like *elastic pricing* allow the system to respect concurrent logins: you can have as many simultaneous login attempts as you like, but each will require another deposit and the price can be adjusted accordingly. There is, therefore, a financial disincentive for a population-based guessing attacker.

Our proposal assumes there is a workable payment system that is real-time, private, widely-accepted, and free of transaction fees. There are multiple proposals [28, 32, 8] that can be the foundation to implement the deposit-based rate-limiting with blockchain technology and smart contracts. By allowing transactions off-blockchain with the confidence of on-blockchain enforceability, they feature instant payments, scalability, and suitable transaction fees. However, a broad adoption of such systems remains a deployment challenge.

4.2 Pricing

Our fundamental aim is to increase the trawling attacker's cost while imposing practically zero additional net cost to the honest user. The advantage of a financial engineering approach is that the degree of protection can be calibrated according to the account's value. Pricing should account for both user's and attacker's perception of value, while volatility should account for attacker activities.

Stolen accounts have a definite resale value on underground Darknet sites. As expected, the asking price varies based on the type of account [37] and many other factors. At one extreme, 160 million LinkedIn account credentials may be purchased for a few thousand dollars [15]. Beyond this direct resale market, the attacker could use an account as part of a larger scheme to steal financial assets, to facilitate the takeover of other accounts (such as email used for fallback authentication).

For the user, accounts have other intangible value. Especially on social networks, these represent an online reputation with relationships that could be harmed. As social networks are now commonly also used by corporate marketing departments, the attacker could damage a corporate brand.

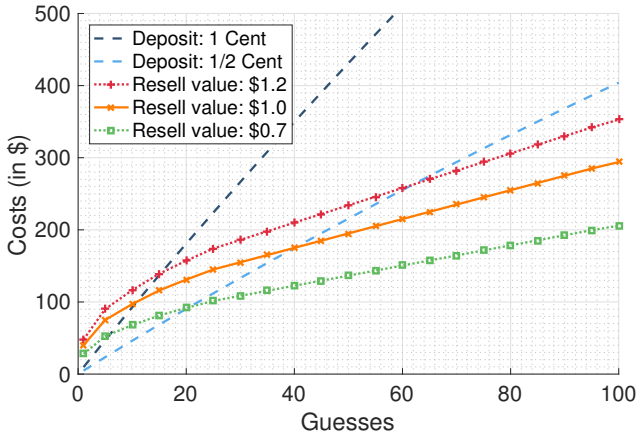


Figure 2: *Static* price simulation considering a *trawling* attacker with perfect knowledge of the distribution. We simulated an attack scenario against 1 000 accounts with two deposit prices (1 and 1/2 cent), and increasing resell values between $0.70 and $1.20.

Users commonly also have an emotional attachment to their online accounts as a result of the investment of time.

We prioritize black market value, as it gives a range of concrete financial values for stolen accounts. Our aim, then, is to increase the attacker's cost of goods sold.

First, we consider a *static pricing* for deposits. Suppose the attacker needs n guesses on average to take over an account with resale value r. To eliminate the profit potential, and ignoring transaction costs, we set the deposit per guess d so that $nd \geq r$. The resale value will vary over time and by account, so for higher-value accounts, d will need adjustments as new data arises.

We simulated an online trawling attacker with perfect knowledge of the password distribution (guessing only correct passwords in the perfect order) against 1 000 accounts. The simulation samples 1 000 passwords at random from a top 10 000 most commonly used password distribution [1]. Our sample

[1]Our approximation of the top 10 000 most common passwords is derived from a weighted combination of the RockYou (32 M), 000Webhost (15 M), and LinkedIn (163 M accounts) password leak.

of 1 000 passwords consists of 806 unique passwords. Note, real-world attackers can only approximate the correct distribution (which is influenced by different factors). Thus, we provide a lower bound on the security offered by the rate-limiting mechanism. Real-world account values differ from one service to another and can only be described in price ranges, e. g., a Gmail account is reported to be worth between $0.70 and $1.20 [37]. We tested two different *static* deposit pricing values, namely 1/2 cent and 1 cent per login attempt. As one can see in Figure 2 even if the highest resell value of $1.20 and a deposit of 1 (1/2) cent is assumed, no more than 15 (60) guesses can be made before the guessing attack becomes uneconomic.

So far we have discussed *static pricing*, where the deposit value is fixed system-wide. One can also consider basing the deposit price on the value/risk of the individual account, with the obvious questions how to rate the value or risk of the account, how to treat changing values of the account, and more. Another condition for *dynamic prices* is past login behavior, and specifically, the number of past failed logins (as a rough estimate for the risk of the current login). This would hardly affect an honest user making a typo once in a while, but increase cost for a guessing adversary. However, some potential problems arise from this dynamic pricing, as it may incentivize phishing attacks (where the phisher could require a high deposit), and may be used for a denial of service attack.

Another design choice we made is refunding the deposit for the current login only; one could additionally refund all deposits for previous failed attempts. This would have two major effects: A guessing attack that is eventually successful would be without cost (but the typically larger share of unsuccessful attacks still carry cost), and an honest user making a typo would ultimately not pay for the typo. Additionally, the honest user would financially profit from unsuccessful attacks. Similar to authentication systems that only allow a specific number of authentication attempts, one might implement partial refunds, e.g., refunding the last 3–5 failed login attempts only. We didn't further pursue these questions for this initial report to simplify the analysis.

4.3 Security and Usability

Login interfaces of major consumer websites such as Google, Microsoft, and Yahoo separate username and password fields onto different pages. This change facilitates multi-step authentication and permits the use of new solutions that complement traditional passwords [11], creates a more consistent sign-in process between desktop computers and mobile phones, makes the sign-in process faster [19], enables sign-in-classification [17], and allows implementation of federated identity schemes like OpenID Connect (OIDC) [40]. The proposed rate-limiting system would add a step to integrate the browsing session with a micropayment transaction and could also be included in federated identity solutions. The user's micropayment wallet will prompt for a payment approval. While a detailed discussion on pseudonymity and blockchain-based cryptocurrencies is beyond the scope of this paper, the Brave browser [12] demonstrates bitcoin-based micropayments integrated into a browser.

Usability issues and attacks on micropayment wallets [27] are a general threat against our idea. However, we would expect that a tightly integrated micropayment solution specifically adapted for the application scenario can overcome these usability problems.

A conceptual problem is that phishing attacks may now even get financially incentivized. We conjecture that for the very moderate deposits around 1 cent, which we propose to use, the overall cost of setting up and maintaining the attack is too high to justify the attack. Furthermore, the pseudonymity of many suitable payment schemes increases the risk for the phisher to get caught by tracing the received payments. (Additional care must, however, be taken when requiring higher deposits or using dynamic pricing.)

Adding another requirement to the login process may decrease usability. However, at the same time we expect other security measures to be eased, e. g., a service can disable CAPTCHA solving for opted-in accounts, thus remove features that do not add any security but negatively affect the user experience [26, 9]. Also, the system would only be active after explicit user consent. To prevent abuse of the fallback authentication technique used, it needs to be as secure as the primary authentication in its resistance to guessing and phishing attacks.

We encourage to minimize the number of unsuccessful logins by the user caused by mistakes, not to inspire to pick simpler passwords and promote password reuse as a consequence. Therefore, a system that securely corrects common typographical errors on behalf of the user [6] and other mechanisms like an option to display the password in plain text [30] could be implemented. Also, the deposit value could be reduced if the user picks a more complex password or adopts other security-conscious behaviors.

5. CONCLUSION

This paper presents a new deposit-based rate-limiting approach seeking to thwart trawling online guessing attacks. While the underlying concept is easy to follow, there are many security and usability as well as payment system-related implications that need to be considered and addressed in more detail. Once the requirements for a suitable payment system can be fulfilled, we recommend further investigations, simulations, and user studies to test the approach in the real-world.

6. REFERENCES

[1] M. Abadi, A. Birrell, M. Burrows, F. Dabek, and T. Wobber. Bankable Postage for Network Services. In *Asian Computing Science Conference*, ASIAN '03, pages 72–90, Mumbai, India, Dec. 2003. Springer.

[2] M. Alsaleh, M. Mannan, and P. C. van Oorschot. Revisiting Defenses against Large-Scale Online Password Guessing Attacks. *IEEE Transactions on Dependable and Secure Computing*, 9(1):128–141, Jan. 2012.

[3] J. Bonneau, C. Herley, P. C. van Oorschot, and F. Stajano. The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes. In *IEEE Symposium on Security and Privacy*, SP '12, pages 553–567, San Jose, California, USA, May 2012. IEEE Computer Society.

[4] J. Bonneau, M. Just, and G. Matthews. What's in a Name? Evaluating Statistical Attacks on Personal Knowledge Questions. In *Financial Cryptography and*

Data Security, FC '10, pages 98–113, Tenerife, Canary Islands, Spain, Jan. 2010. Springer.

[5] W. E. Burr, D. F. Dodson, and W. T. Polk. Electronic Authentication Guideline: NIST Special Publication 800-63-2. Technical report, National Institute of Standards & Technology, 2013.

[6] R. Chatterjee, A. Athalye, D. Akhawe, A. Juels, and T. Ristenpart. pASSWORD tYPOS and How to Correct Them Securely. In *IEEE Symposium on Security and Privacy*, SP '16, pages 799–818, San Jose, California, USA, May 2016. IEEE Computer Society.

[7] S. Chiasson, P. C. van Oorschot, and R. Biddle. A Usability Study and Critique of Two Password Managers. In *USENIX Security Symposium*, SSYM '06, pages 1–16, Vancouver, B.C., Canada, July 2006. USENIX Association.

[8] E. Community and Contributors. A Next-Generation Smart Contract and Decentralized Application Platform. Technical report, `https://github.com/ethereum/wiki/wiki/White-Paper`, as of June 19, 2017, Sept. 2014.

[9] E. De Cristofaro, H. Du, J. Freudiger, and G. Norcie. A Comparative Usability Study of Two-Factor Authentication. In *Workshop on Usable Security*, USEC '14, San Diego, California, USA, Feb. 2014. The Internet Society.

[10] S. Dritsas, Y. Soupionis, M. Theoharidou, Y. Mallios, and D. Gritzalis. SPIT Identification Criteria Implementation: Effectiveness and Lessons Learned. In *International Information Security Conference*, IFIP SEC '08, pages 381–395, Milano, Italy, Sept. 2008. Springer.

[11] J. Esparza. Minor Updates to Your Google Sign-in Experience, May 2015. `https://productforums.google.com/forum/#!topic/gmail/oAsE-6wmaSU`, as of June 19, 2017.

[12] K. Finley. The Creator of JavaScript is Out to Upend the Ad Industry, Jan. 2016. `https://www.wired.com/2016/01/the-creator-of-javascript-wants-to-blow-up-the-ad-industry/`, as of June 19, 2017.

[13] D. Florêncio and C. Herley. Is Everything We Know about Password Stealing Wrong? *IEEE Security Privacy*, 10(6):63–69, Nov. 2012.

[14] D. Florêncio, C. Herley, and P. C. van Oorschot. An Administrator's Guide to Internet Password Research. In *Large Installation System Administration Conference*, LISA '14, pages 44–61, Seattle, Washington, USA, Nov. 2014. USENIX Association.

[15] L. Franceschi-Bicchierai. Another Day, Another Hack: 117 Million LinkedIn Emails And Passwords, May 2016. `https://motherboard.vice.com/en_us/article/another-day-another-hack-117-million-linkedin-emails-and-password`, as of June 19, 2017.

[16] L. Franceschi-Bicchierai. The Worst Hacks of 2016, Dec. 2016. `https://motherboard.vice.com/en_us/article/the-worst-hacks-of-2016`, as of June 19, 2017.

[17] D. M. Freeman, S. Jain, M. Dürmuth, B. Biggio, and G. Giacinto. Who Are You? A Statistical Approach to Measuring User Authenticity. In *Symposium on Network and Distributed System Security*, NDSS '16, San Diego, California, USA, Feb. 2016. The Internet Society.

[18] Google, Inc. 2-Step Verification, Dec. 2016. `https://www.google.com/landing/2step/`, as of June 19, 2017.

[19] Google, Inc. Learn About the New Google Sign-in Page, May 2017. `https://support.google.com/accounts/answer/7338427`, as of June 19, 2017.

[20] P. A. Grassi, J. L. Fenton, and W. E. Burr. Digital Identity Guidelines - Authentication and Lifecycle Management: Draft NIST Special Publication 800-63-B. Technical report, National Institute of Standards & Technology, 2017.

[21] V. Griffith and M. Jakobsson. Messin' with Texas: Deriving Mother's Maiden Names Using Public Records. In *Applied Cryptography and Network Security*, ACNS '05, pages 91–103, New York, New York, USA, June 2005. Springer.

[22] H. Habib, J. Colnago, W. Melicher, B. Ur, S. Segreti, L. Bauer, N. Christin, and L. Cranor. Password Creation in the Presence of Blacklists. In *Workshop on Usable Security*, USEC '17, San Diego, California, USA, Feb. 2017. Internet Society.

[23] C. Herley and D. Florêncio. Protecting Financial Institutions from Brute-Force Attacks. In *International Information Security Conference*, IFIP SEC '08, pages 681–685, Milano, Italy, Sept. 2008. Springer.

[24] J. C. Hull. *Options, Futures and Other Derivatives*. Pearson, 9 edition, 2014.

[25] T. Hunt. Password reuse, credential stuffing and another billion records in Have I been pwned, May 2017. `https://www.troyhunt.com/password-reuse-credential-stuffing-and-another-1-billion-records-in-have-i-been-pwned/`, as of June 19, 2017.

[26] K. Krol, E. Philippou, E. De Cristofaro, and M. A. Sasse. "They brought in the horrible key ring thing!" Analysing the Usability of Two-Factor Authentication in UK Online Banking. In *Symposium on Network and Distributed System Security*, NDSS '15, San Diego, California, USA, Feb. 2015. The Internet Society.

[27] K. Krombholz, A. Judmayer, M. Gusenbauer, and E. Weippl. The Other Side of the Coin: User Experiences with Bitcoin Security and Privacy. In *Financial Cryptography and Data Security*, FC '16, pages 555–580, Christ Church, Barbados, Feb. 2016. Springer.

[28] R. Kumaresan, T. Moran, and I. Bentov. How to Use Bitcoin to Play Decentralized Poker. In *ACM Conference on Computer and Communications Security*, CCS '15, pages 195–206, Denver, Colorado, USA, Oct. 2015. ACM.

[29] Z. Li, W. He, D. Akhawe, and D. Song. The Emperor's New Password Manager: Security Analysis of Web-based Password Managers. In *USENIX Security Symposium*, SSYM '14, pages 465–479, San Diego, California, USA, Aug. 2014. USENIX Association.

[30] W. Melicher, D. Kurilova, S. M. Segreti, P. Kalvani, R. Shay, B. Ur, L. Bauer, N. Christin, L. F. Cranor, and M. L. Mazurek. Usability and Security of Text Passwords on Mobile Devices. In *ACM SIGCHI*

Conference on Human Factors in Computing Systems, CHI '16, pages 527–539, Santa Clara, California, USA, May 2016. ACM Press.

[31] M. Motoyama, K. Levchenko, C. Kanich, D. McCoy, G. M. Voelker, and S. Savage. Re: CAPTCHAs: Understanding CAPTCHA-solving Services in an Economic Context. In *USENIX Security Symposium*, SSYM '10, pages 435–452, Washington, D.C., USA, Aug. 2010. USENIX Association.

[32] J. Poon and T. Dryja. The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments. Technical report, `https://lightning.network/lightning-network-paper.pdf`, as of June 19, 2017, Jan. 2016.

[33] J. Rosenberg and C. Jennings. The Session Initiation Protocol (SIP) and Spam. RFC 5039, RFC Editor, Jan. 2008.

[34] D. Rosenblum. What Anyone Can Know: The Privacy Risks of Social Networking Sites. *IEEE Security and Privacy*, 5(3):40–49, June 2007.

[35] S. Schechter, A. J. B. Brush, and S. Egelman. It's No Secret. Measuring the Security and Reliability of Authentication via "Secret" Questions. In *IEEE Symposium on Security and Privacy*, SP '09, pages 375–390, Oakland, California, USA, May 2009. IEEE Computer Society.

[36] S. Schechter, C. Herley, and M. Mitzenmacher. Popularity Is Everything: A New Approach to Protecting Passwords from Statistical-Guessing Attacks. In *USENIX Workshop on Hot Topics in Security*, HotSec '10, Washington, District of Columbia, USA, Aug. 2010. USENIX Association.

[37] M. Stockley. What your hacked account is worth on the Dark Web, Aug. 2016. `https://nakedsecurity.sophos.com/2016/08/09/what-your-hacked-account-is-worth-on-the-dark-web/`, as of June 19, 2017.

[38] N. Szabo. Formalizing and Securing Relationships on Public Networks. *First Monday*, 2(9), 1997.

[39] M. Tracy, W. Jansen, K. A. Scarfone, and T. Winograd. Guidelines on Securing Public Web Servers: NIST Special Publication 800-44-2. Technical report, National Institute of Standards & Technology, 2007.

[40] A. Vapen, N. Carlsson, A. Mahanti, and N. Shahmehri. A Look at the Third-Party Identity Management Landscape. *IEEE Internet Computing*, 20(2):18–25, Mar. 2016.

[41] D. Wang, Z. Zhang, P. Wang, J. Yan, and X. Huang. Targeted Online Password Guessing: An Underestimated Threat. In *ACM Conference on Computer and Communications Security*, CCS '16, pages 1242–1254, Vienna, Austria, Oct. 2016. ACM Press.

[42] L. Zhao and M. Mannan. Explicit Authentication Response Considered Harmful. In *New Security Paradigms Workshop*, NSPW '13, pages 77–86, The Banff Centre, Banff, Canada, Sept. 2013. ACM Press.