Global Journal of Computer Sciences: Theory and Research

Volume 07, Issue 2, (2017) 92-101

www.gjcs.eu

The implementation of two-factor web authentication system based on facial recognition

Sultan Zavrak*, Department of Computer Engineering, Duzce University, Duzce 81620, Turkey.
Seyhmus Yilmaz, Department of Computer Engineering, Duzce University, Duzce 81620, Turkey.
Huseyin Bodur, Department of Computer Engineering, Duzce University, Duzce 81620, Turkey.
Sinan Toklu, Department of Computer Engineering, Duzce University, Duzce 81620, Turkey.

Suggested Citation:
Zavrak,S., Yilmaz, S., Bodur, H. & Toklu, S. (2017). The implementation of two-factor web authentication system based on facial recognition. *Global Journal of Computer Sciences: Theory and Research. 7*(2), 92-101.

Received April 8, 2017; revised June 21, 2017; accepted August 8, 2017.
Selection and peer review under responsibility of Prof. Dr. Dogan Ibrahim, Near East University, North Cyprus.

Abstract

The security of the web is a very important issue, because every day we make a variety of operations in it, for different reasons, during the day. Apart from protecting the information, contacts, accounts and data on the web, such data should be inaccessible to third-party persons. This in turn depends on the success of the authentication process performed on the individual web. With authentication, it is possible for users to protect their information and make their transactions only for themselves. However, the authentication mechanism used at this point must have a high level of safety. With the purpose to damage a person's privacy and access account information and gain profit in this way, many malicious persons have developed various methods of attacks to bypass authentication mechanisms. These methods sometimes succeed on a variety of authentication mechanisms, and put users and relevant websites into a difficult situation, and may even damage them in a variety of aspects. In order to protect personal information on the web system and provide the security of transactions carried out at a high level, in this study, we propose a two-factor authentication mechanism based on facial recognition. Besides, we discuss some implementation details about the proposed method. The proposed method aims to bring a new approach to the authentication system to perform our online process with the highest security. In addition to the standard authentication systems, using face recognition as a secondary level of security will contribute to the emergence of a new authentication mechanism.

Keywords: Web authentication, two-factor authentication, web security, facial recognition.

***** ADDRESS FOR CORRESPONDENCE: **Sultan Zavrak,** Duzce University, Engineering Faculty B Blok Floor: 3/318: Sultan, Zavrak, Duzce University, Duzce, Turkey. *E-mail address*: sultanzavrak@duzce.edu.tr / Tel.: +90-380-542-1036/4730

1. Introduction

The Web has become a very important factor for carrying out the daily business of people in a company network or on the Internet. People do many jobs, such as accessing their mails, accessing their financial accounts, paying public bills, shopping online, getting electronic health records and so on, through their own computers via a web browser. Web authentication is the primary defense line for everyone to protect their web accounts and ensure data security. Generally, a user authenticates his or her own username and password for a web application hosted on a remote server by itself (either manually or automatically via a password manager) on the login page of the application itself. The password is the actual method for web authentication [1]. However, it cannot provide sufficient protection for password authentication only, because the mechanism is prone to many attacks such as shoulder surfing attack [2], brute force password guessing attack [3–5], man in the middle (MITM) attack [6] and phishing attack [7, 8].

Web browsers like Chrome, Firefox and Internet Explorer, which use a built-in password manager, have been developed to increase the identity security on the web and simplify password management. At this point, independent password managers [9] (e.g., Password and KeePass) and web-based password managers (e.g., LastPass and PasswordBox) running in a web browser have become very popular. However, due to local or remote unsafe computing environments, a stand-alone password manager does not provide a security guarantee at a sufficient level.

Zhao and Yue [10] show that none of the browsers with built-in password managers on the main web browsers can prevent malware from stealing passwords in a computer environment. Moreover, in recent years works that have been done on web password autoloading [11] and web-based password management systems [12] have revealed serious security vulnerabilities that can be misused for password attacks in popular password managers.

Recently, data breaches and password database leaks have been witnessed frequently in popular websites such as LinkedIn [13], Yahoo [14] and Gmail [15]. These password spoofs threaten not only the data security of millions of people on those sites, but also the security of other websites, because users reuse the same passwords on other websites [16]. At the point of making this problem worse, attackers are applying MITM and phishing attacks to capture users' passwords. The recent MITM attack against Google users in Iran [6] shows that even reinforced and compromised websites may be exposed to MITM attacks. According to [17], the worldwide number of websites exposed to total phishing attacks in the first quarter of 2014 has increased by 10.7% to 125,215, according to figures in the fourth quarter of 2013. Although TLS/SSL protocols can be applied against MITM and phishing attacks, the security provided by HTTP over TLS/SSL (HTTPS) depends on the validity of the certificate [18], so the actual implementation [19, 20] is generally weak. In addition, HTTPS is not available on many sites, mainly some government websites (e.g. www.basbakanlik.gov.tr and www.adalet.gov.tr).

Two-factor authentication (TFA) is strongly recommended and encouraged to increase web identity security, as password-only authentication is obviously insufficient. At this point, although special hardware-based TFA solutions (e.g., SecurID and smart card) have been introduced long ago, they have not been widely used yet. With the advancement of mobile computing technologies in the past decade, TFA systems supported by many mobile devices have been proposed [7, 21–23]; in addition to encryption, reliable mobile device support has become a secondary factor. TFA systems have been developed that are used in SMS-based (e.g., [24, 25]) and software-based (e.g., [26]) mobile phones, especially smartphones.

TFA requires two or more verification factors to be presented. Examples of these factors are a password known only by the user, a secure token that the user has and a biometric feature of only one user. At this point, using more than one factor usually provides a higher level of authentication assurance. In RSA SecurID, for example, biometric features such as fingerprints or one-time passwords (OTPs), passwords are combined with security tokens.

In this study, a new web interface TFA mechanism that is resilient to attacks such as MITM and phishing in user login systems and works on face recognition based and mobile devices is proposed. In addition, a web application prototype with a user interface that supports the suggested method, and a prototype of a mobile application that can perform second-level authentication using face recognition and integrate with the web application have been developed and the applicability of the proposed mechanism has been discussed based on certain parameters. As a result, the proposed mechanism seems to be resistant to MITM and punctuation attacks.

This paper is organized as follows. In Section 2, we summarise the literature studies. In Section 3, the proposed system and the implementation details of software prototypes are explained. In Section 4, the evaluation of the system is discussed. In the last section, the concluding remarks are stated.

2. Related Work

With the widespread usage of mobile phones, authentication tools have also been updated to fit the mobile structure. In this way, the short message service (SMS) [24] or an interactive telephone conversation [25] or a mobile device application can be used to convert the computer user's mobile device into a secure token device.

Mobile-assisted authentication schemes [7, 22, 23, 27] have been proposed to protect the user from stealing the password on an insecure computer or from phishing attacks. In these diagrams, it is assumed that mobile devices are reliable and capable of performing certain computer operations such as hashing.

Phoolproof [7] is an open-key based scheme used to strengthen the bank transaction system. According to the diagram, after the mobile user selects a trusted bank site, it is necessary for the user to wait for data exchange between the mobile device, which is a secure token device, and the computer. This data exchange includes security mechanisms that protect the system against attacks at higher levels. MP-Auth [22] is a scheme that defends against keylogger and phishing attacks by means of a mobile device by triggering the corresponding security methods to re-encrypt the username and password entered by the user. Both Phoolproof and MP-Auth require a wireless connection and a well-implemented SSL/TLS certificate.

Czeskis *et al.* [23] proposed an intelligent device-based authentication scheme called PhoneAuth, to enhance user security within the authentication scheme. Recently, image-based communication has attracted considerable attention, along with the growing popularity of mobile devices where more than one camera is involved.

McCune *et al.* [28] proposed an authentication scheme called a Seeing-is-Believing (SiB) that utilises a one-way visual channel between a 2D barcode and a camcorder mobile device.

Saxena *et al.* [29] proposed a short-range device mapping protocol based on a one-way visual channel, called visual authentication based on integrity checking (VIC). Another wireless communication channel (such as Bluetooth) must be used to complete the pairing process. Neither SiB nor VIC is well suited for authentication.

Xie *et al.* [30] proposed CamTalk, a light-based communication scheme for bidirectional secure data transfer between smart devices using smart device screen camera channel.

Recently, Xie *et al.* [31] proposed CamAuth, a web authentication mechanism against a variety of password attacks, such as phishing, by exploiting popular mobile devices and digital cameras. In CamAuth, the mobile device is used as a second authentication factor to authenticate the identity of the person performing the web entry from the personal computer. CamAuth uses public key cryptography to provide authentication process security. A major drawback of the proposed mechanism is that it is not biometric-based, and at the same time requires a camera in the personal computer.

3. Two-Factor Authentication Mechanism Based on Face Recognition

3.1. System Design

With the proposed system, it is aimed to guarantee authentication security in an efficient and appropriate way at the entries made through the web browser of a personal computer. At this point, the system uses the mobile device as a reliable secondary authentication factor.

Figure 1 shows a diagram of the normal authentication process. This process consists of interactions between four different entities: user, personal computer, mobile device and web server.

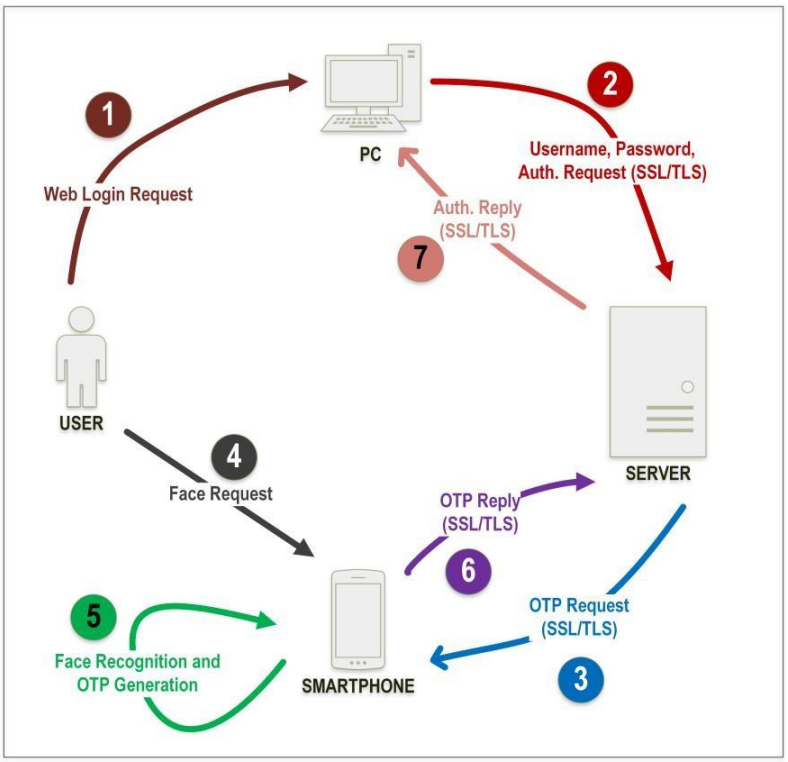

Figure 1. Proposed authentication diagram

The process steps in Figure 1 can be summarized as follows:

1. The user starts login process by entering the user name and password manually or automatically.
2. The web browser activates username and password (or the hash value of the password; 'password' is used for display in Figure 1) to send from a secure connection. In the meantime, the web browser goes through the validation phase and waits for authentication to be completed by the server.
3. The web server requires an OTP from the smartphone and the user is expected to authenticate from the mobile device.
4. The requesting of OTP by server triggers the user to show the face to the application.
5. If the user shows his/her face and the face recognition process is successfully completed, OTP is generated. If facial recognition fails, OTP is not produced.
6. The generated OTP is notified to the server via an encrypted connection.
7. If a valid OTP is generated (face recognition is successful), the server notices the authentication successfully to the web browser and the user login is terminated successfully. If an invalid OTP is generated, the web browser is notified that the authentication failed and the login process is terminated.

Figure 2 shows a diagram of the process of activating the mobile device, in other words adding the device as a trusted device. This process also means enabling TFA login. This process consists of interactions between three different entities. These are the user, mobile device and web server.

The process steps in Figure 2 can be summarized as follows:

1. The user initiates the mobile device registration process by manually entering the username, password and phone number.
2. The mobile device sends the user name, phone number and password data (or the summary of the hash value of the password; 'password' is used for display in Figure 1) from the secure connection to the server.
3. The server generates an OTP to the specified phone number in case the user name and password data are correct and sends it as an SMS.
4. If the OTP that comes from the server is verified, the user is expected to show his/her face for face registration.
5. If the user shows his/her face and face recognition is successfully completed, the face of the user is registered on the mobile device. If face detection fails, device registration will be invalid.
6. If the face registration process is successfully completed, the mobile device ID number is generated and this number is recorded in the server.

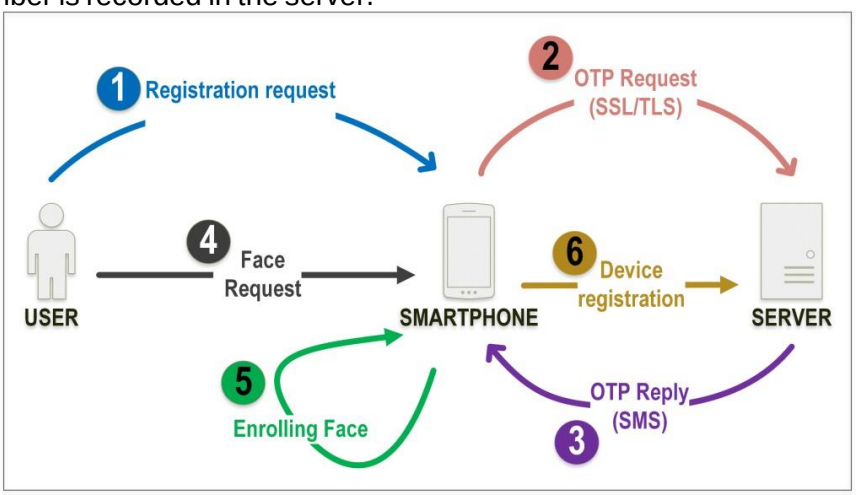

Figure 2. Mobile device registration (activation) diagram

In this work, it is assumed that some features are present in the devices to be used. The first is an Internet connection between the personal computer and the mobile device and the web server. The second is the presence of a camera (the camera feature is now available on almost all phones on the market today) that will be used for face recognition on the mobile device. The third is that HTTPS, which ensures that the connection is completely secure, will be used for web authentication. Our fourth and last hypothesis is that the user will be able to do the web login process on more than one personal computer, and will always be able to use the same mobile device for authentication.

An important part of the proposed mechanism is the production and use of OTP. Many studies [32–35] have been published in the literature for OTP generation. In this work, Liu and Zhang's OTP scheme [35], which is resilient to some attacks such as phishing, impersonation is used.

3.2. Prototype implementation

The implementation of the proposed mechanism software prototype consists of two parts: web application and mobile application. The PHP programming language [36], MySQL database [37] and JavaScript have been used because of their popularity in web application. The web application satisfies

the user with a web page, as shown in Figure 3, where a user name and password data can be retrieved for proper operation of the proposed mechanism. Then, if the username and password are entered correctly, a notification including OTP request for login process is sent to the mobile application and OTP response is expected for authentication from the mobile application. The OTP waiting web page is as shown in Figure 4. The Google Firebase API [38] has been used due to its robust infrastructure and optimized Android operating system performance to send a notification.

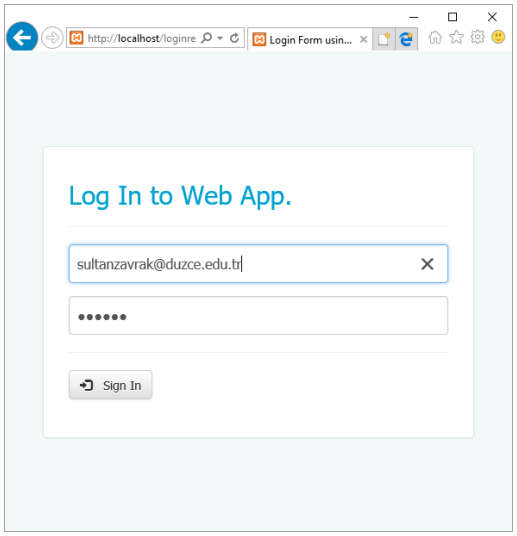

Figure 3. The login web page

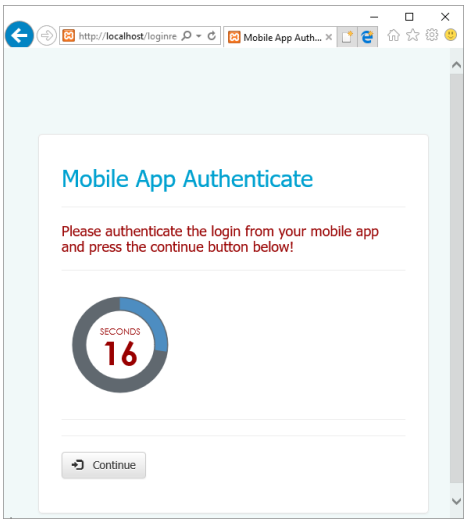

Figure 4. Authentication waiting web page

Figure 5. Mobile authentication application—notifications interface

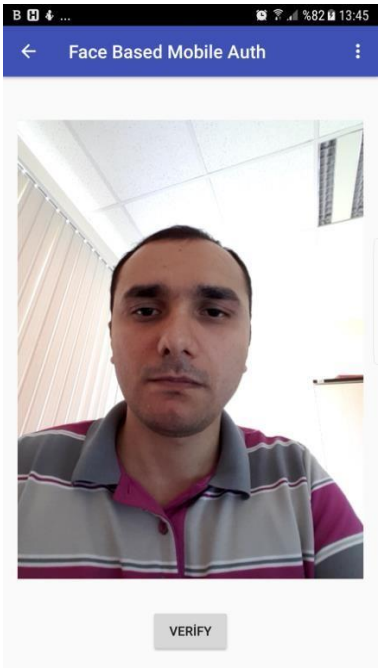

Figure 6. Mobile authentication application—face verification interface

Mobile Authentication prototype was implemented as an Android 7 application. The application is responsible for registering and authenticating the mobile device. The application uses the Google Firebase infrastructure to receive notifications from the web. In addition, the VeriLook SDK [39] is used for face registration and face recognition. The mobile application screenshot showing web login request notifications is shown in Figure 5, the screenshot in which the face verification process is performed is shown in Figure 6 and if the face verification is successful the screenshot of the status message is shown in Figure 7.

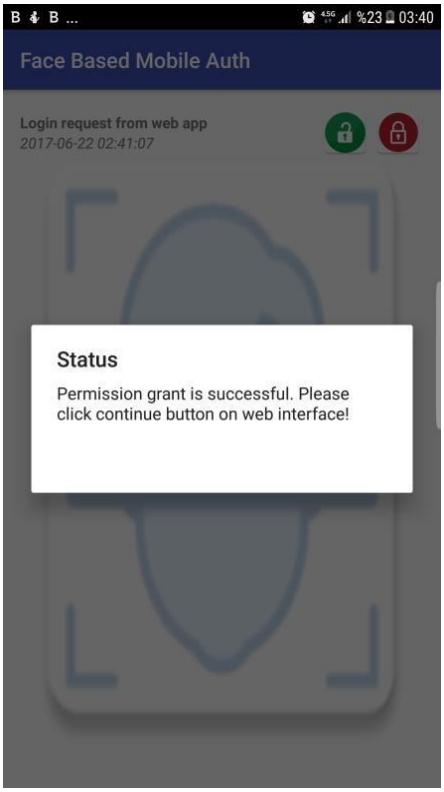

Figure 7. Mobile authentication application—permission granted message

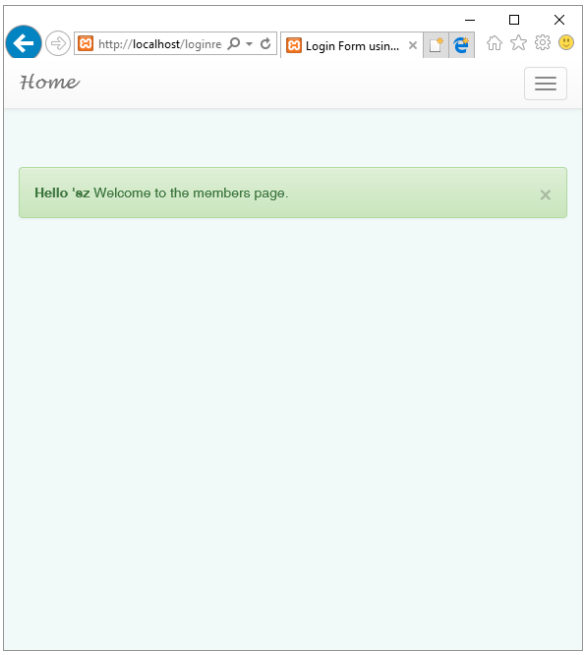

Figure 8. Web application—protected area

4. Evaluation

The evaluation of the proposed scheme is performed by using the web authentication evaluation framework recommended by Bonneau [1]. The recommended scheme is compared to the most popular TFA scheme, Google 2-step verification (2SV) [27] and CamAuth, which is a TFA scheme with a corresponding mobile device base and ciphers. The comparison results are shown in Table 1. The proposed scheme is similar to 2SV and CamAuth in terms of usability. Face recognition has already begun to be widely used using smartphones. According to the definition of these advantages [1], it is considered to be easy-to-learn and easy-to-use. We believe that the proposed scheme and CamAuth (and 2SV) are at the same level for easy-recovery-from-loss, because the rescue mechanisms are very similar, despite the fact that they both have the same pinpoint: Users need to cancel the old device and install the application on the new device. Then open your device and save the new device.

Table 1. The recommended mechanism is to compare CamAuth, Google 2-step verification and cipher mechanisms (Note: 'y' is provided as a benefit, 's' as a bit)

	Usability													**Deployability**					**Security**						
Scheme	Scalable-for-users	Nothing-to-carry	Quasi-nothing-to-carry	Easy-to-learn	Easy-to-use	Infrequent-errors	Easy-recovery-from-loss	Accessible	Negligible-cost-per-user	Server-compatible	Browser-Compatible	Mature	Non-proprietary	Resilient-to-physical-observation	Resilient-to-targeted-impersonation	Resilient-to-throttled-guessing	Resilient-to-Unthrottled-guessing	Resilient-to-internal-observation	Resilient-to-leaks-from-other-Verifiers	Resilient-to-phishing	Resilient-to-theft	No-trusted-third-party	Requiring-explicit-consent	Unlinkable
Passwords		y	y	y	y	s	y	y	y	y	y	y	y		s						y	y	y	y
Google 2-step verification			y	y	s	s	s	s			y	y			s	y			y	y	y	y	y	y
CamAuth		y	y	s	s	s	s	s	s	s	s		y	y	y	y	y	s	y	y	y	y	y	y
Our scheme		y	y	y	y	s	s	s	s	s	y		y	y	y	y	y	s	y	y	y	y	y	y

For evaluation of the deployability of the mechanism we propose, the distribution of the application is usually based on what changes are needed in the existing systems. Our scheme is designed to be applicable at the user level and application layer. Distributability is closely comparable to CamAuth, as it does not require any changes to the OS kernel, device driver or sublayer protocols. The proposed scheme is safe to guess about the security, physical observations and impersonation of the target identity, whether the attacker cannot log in even though the user still has the password without the device. The device and the computer have to be put in danger by malicious software. This percentage can be quite flexible against internal observation. Since the device has a separate key pair (i.e., verifier) for each web application, the proposed scheme is resistant to leakage from other verifiers. It is absolutely resistant to phishing and theft because of its two-step authentication.

The performance of our proposed scheme, that is, the time spent inputting, certainly influences the user experience. We are interested in performance because our scheme includes mobile device face recognition in an entry. In our developed mobile app, we performed an experiment to measure the average duration of an average user's session. We used a Samsung S7 smartphone with a 5-megapixel front camera for the test of the developed mobile application. Five users joined the test process and each performed ten web logins. The smartphone spent an average of 3.4 seconds to launch the application, receive notifications, make face recognition and present. Face recognition is an average of 2.4 seconds after the notification of the request to enter the mobile device.

5. Discussion and Conclusion

In this study, an application that uses smartphones with a very high market share as a secondary factor has been realized. In this application, a new two-step authentication mechanism (scheme) is proposed, which uses the cameras that come integrated with these devices for face recognition purposes. This application has developed a web application prototype and a mobile application prototype, if necessary. The proposed mechanism can work correctly and steadily without any modifications to the existing network protocols and the operating system of the smartphone and

personal computer. In addition, it effectively eliminates password stealing attacks such as MITM attacks and phishing attacks. The developed prototype system and the initial user experience demonstrate the applicability of the mechanism.

Acknowledgements

This study was supported by the Duzce University Research Fund Project Number: 2016.06.01.474.

References

[1] J. Bonneau *et al.*, "The quest to replace passwords: a framework for comparative evaluation of web authentication schemes," in: *2012 IEEE Symposium on Security and Privacy*, 2012, pp. 553–567.
[2] F. Tari *et al.*, "A comparison of perceived and real shoulder-surfing risks between alphanumeric and graphical passwords," in: *Proceedings of the 2nd Symposium on Usable Privacy and Security*, 2006, pp. 56–66.
[3] D. C. Feldmeier and P. R. Karn, "Unix password security-ten years later," in: *Conference on the Theory and Application of Cryptology*, 1989, pp. 44–63.
[4] A. Narayanan and V. Shmatikov, "Fast dictionary attacks on passwords using time-space tradeoff," in: *Proceedings of the 12th ACM Conference on Computer and Communications Security*, 2005, pp. 364–372.
[5] J. Bonneau, "The science of guessing: analyzing an anonymized corpus of 70 million passwords," in: *2012 IEEE Symposium on Security and Privacy (SP)*, 2012, pp. 538–552.
[6] S. Sengupta, "In latest breach, hackers impersonate Google to snoop on users in Iran," 2011. *Online+. Available: http://www.nytimes.com/2011/08/31/technology/internet/hackers-impersonate-google-to-snoop-on-users-in-iran.html?_r=0. Accessed June 12, 2016.
[7] B. Parno *et al.*, "Phoolproof phishing prevention," in: *International Conference on Financial Cryptography and Data Security*, 2006, pp. 1–19.
[8] C. Yue and H. Wang, "BogusBiter: a transparent protection against phishing attacks," in: *ACM Trans. Internet Technol.*, vol. 10, issue 2, p. 6, 2010.
[9] K.-P. Yee and K. Sitaker, "Passpet: convenient password management and phishing protection," in: *Proceedings of the 2nd Symposium on Usable Privacy and Security*, 2006, pp. 32–43.
[10] R. Zhao and C. Yue, "All your browser-saved passwords could belong to us: a security analysis and a cloud-based new design," in: *Proceedings of the 3rd ACM Conference on Data and Application Security and Privacy*, 2013, pp. 333–340.
[11] D. Silver *et al.*, "Password managers: attacks and defenses," in: *Usenix Security*, 2014, pp. 449–464.
[12] Z. Li *et al.*, "The emperor's new password manager: security analysis of web-based password managers," in: *USENIX Security*, 2014, pp. 465–479.
[13] Z. Whittaker, "6.46 million LinkedIn passwords leaked online," 2012. *Online+. Available: http://www.zdnet.com/article/6-46-million-linkedin-passwords-leaked-online/. Accessed June 16, 2016.
[14] D. Hamilton, "Yahoo's password leak: What you need to know (FAQ)," 2012.
[15] J. Leyden, "Leak of '5 MEELLLION Gmail passwords' creates security flap," 2014. *Online+. Available: http://www.theregister.co.uk/2014/09/11/gmail_password_leak_flap/. Accessed May 12, 2017.
[16] D. Florencio and C. Herley, "A large-scale study of web password habits," in: *Proceedings of the 16th International Conference on World Wide Web*, 2007, pp. 657–666.
[17] APWG, "Phishing attack trends report: Q1 2014," 2014. *Online+. Available: http://docs.apwg.org/reports/apwg_trends_report_q1_2014.pdf. Accessed May 12, 2016.
[18] S. Schoen and E. Galperin, "Iranian man-in-the-middle attack against Google demonstrates dangerous weakness of certificate authorities," 2011. *Online+. Available: https://www.eff.org/deeplinks/2011/08/iranian-man-middle-attack-against-google. Accessed May 12, 2016.
[19] L. Bershidsky, "Heartbleed's password heartbreak," 2014. *Online+. Available: https://www.bloomberg.com/view/articles/2014-04-11/heartbleed-shows-open-source-needs-your-cash. Accessed May 12, 2016.
[20] M. Riley, "NSA said to have used heartbleed bug, exposing consumers." *Online+. Available: http://www.bloomberg.com/news/articles/2014-04-11/nsa-said-to-have-used-heartbleed-bug-exposing-consumers. Accessed May 12, 2016.

[21] M. Wu *et al.*, "Secure web authentication with mobile phones," in: *DIMACS Workshop on Usable Privacy and Security Software*, 2004, vol. 2010.
[22] M. Mannan and P. C. van Oorschot, "Leveraging personal devices for stronger password authentication from untrusted computers," *J. Comput. Secur.*, vol. 19, issue 4, pp. 703–750, 2011.
[23] A. Czeskis *et al.*, "Strengthening user authentication through opportunistic cryptographic identity assertions," in: *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, 2012, pp. 404–414.
[24] "Mobile-OTP: mobile one time passwords," *Mobile-OTP*, 2016. [Online]. Available: http://motp.sourceforge.net/. Accessed May 12, 2016.
[25] Duo Security, "Duo security: two-factor authentication made easy," 2016. *Online+. Available: https://www.duosecurity.com/. Accessed May 12, 2016.
[26] Google, "Google 2-step verification." *Online+. Available: http://www.google.com/landing/2step/. Accessed May 12, 2016.
[27] D. Balfanz and E. W. Felten, "Hand-held computer scan be better smartcards," 1999.
[28] J. M. McCune *et al.*, "Seeing-is-believing: using camera phones for human-verifiable authentication," in: *IEEE Symposium on Security and Privacy*, 2005, pp. 110–124.
[29] N. Saxena *et al.*, "Secure device pairing based on a visual channel: design and usability study," *IEEE Trans. Inf. Forensics Secur.*, vol. 6, issue 1, pp. 28–38, 2011.
[30] M. Xie *et al.*, "CamTalk: a bidirectional light communications framework for secure communications on smartphones," in: *SecureComm*, 2013, pp. 35–52.
[31] M. Xie *et al.*, "CamAuth: securing web authentication with camera," in: *2015 IEEE 16th International Symposium on High Assurance Systems Engineering (HASE)*, 2015, pp. 232–239.
[32] L. Lamport, "Password authentication with insecure communication," *Commun. ACM*, vol. 24, issue 11, pp. 770–772, 1981.
[33] K. Bicakci and N. Baykal, "Infinite length hash chains and their applications," in: *Proceedings of the 11th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, 2002, pp. 57–61.
[34] M. H. Eldefrawy *et al.*, "Otp-based two-factor authentication using mobile phones," in: *2011 8th International Conference on Information Technology: New Generations*, 2011, pp. 327–331.
[35] H. Liu and Y. Zhang, "An improved one-time password authentication scheme," in: *15th IEEE International Conference on Communication Technology*, 2013, pp. 1–5.
[36] PHP, "PHP: hypertext preprocessor." *Online+. Available: http://php.net/. Accessed June 20, 2017.
[37] MySQL, "MySQL." *Online+. Available: https://www.mysql.com/. Accessed June 20, 2017.
[38] Google, "Firebase." *Online+. Available: https://firebase.google.com/. Accessed June 20, 2017.
[39] NEUROtechnology, "VeriLook face identification technology, algorithm and SDK for PC, smartphones and Web."*Online+. Available: http://www.neurotechnology.com/verilook.html. Accessed June 20, 2017.