Towards Securing PIN-based Authentication on Smartwatches

Meriem Guerar, Luca Verderame, Alessio Merlo
DIBRIS
University of Genova
Genova, Italy
name.surname@unige.it

Mauro Migliardi
DEI
University of Padua
Padua, Italy
mauro.migliardi@unipd.it

***Abstract*—Smartwatches offer new capabilities to develop sophisticated applications that make daily life easier and more convenient for consumers, and are becoming increasingly ubiquitous. The kind of services these devices are capable to provide include applications for mobile payment, ticketing, identification, access control, etc. While this makes modern smartwatches very powerful devices, it also makes them very attractive targets for attackers. PINs and Pattern Lock have been widely used in smartwatches for user authentication, however, those types of passwords are not robust against various forms of attacks, such as side channel, phishing, smudge, shoulder surfing and video recording attacks. In this work, we propose 2GesturePIN, a new authentication method that allows users to authenticate securely to their smartwatches and sensitive services through solely two gestures. It leverages the rotating bezel or the crown which are the most intuitive channels to interact with a smartwatch. 2GesturePIN enhances the resilience of the regular PIN to common attacks while maintaining a high level of usability.**

***Index Terms*—Smartwatch, Authentication, PIN, Bezel, Crown.**

I. Introduction

In recent years, the market for smartwatches has been steadily growing. From being merely an extension of a smartphone, these devices became more independent, and a lot of sensitive information is now stored on them. Besides health and fitness tracking, smartwatches can be used for making payments, identification, ticketing, and controlling access to physical spaces. As such, the protection of smartwatches sensitive data became of paramount importance.

One solution to protect user's sensitive data from malware is through Trusted Execution Environment (TEE) because it ensures that data is stored, processed and protected within a totally trusted environment that malware cannot tamper with.

TEE has been widely deployed in many smartphones and, recently, it has been ported also on smartwatches. For example, Samsung Gear S2 and S3 contain Knox which is a mobile security platform that provides a trusted execution environment based on the ARM TrustZone technology [1].

Unfortunately, the adoption of TEE comes with a security drawback, namely the access to sensitive data stored in the TrustZone is the weakest link for the security of NFC transactions. This is due to the usage of PIN codes to enable the access to sensitive information: such mechanism is vulnerable to several attacks (e.g, side channel, phishing, shoulder surfing and video recording attacks) and can be easily bypassed [2]–[4]. Therefore, such mechanism is currently too unreliable for security-sensitive operations (e.g, Payment using Samsung pay, Apple pay and Android pay) on smartwatches. The adoption of pattern locks as an alternative to PIN does not provide stronger security guarantees [2], [5].

Although TEE offers a trusted user interface (UI) which ensures that malware running on the device cannot steal data displayed or typed on the screen, this is not sufficient to prevent data leakage from other components of the device that interact with the TEE (e.g., accelerometer, gyroscope, and orientation sensors), as they allow to carry out side channel attacks [6]. Another security issue is that although the TEE locks the screen whenever a trusted application wants to use it, the user cannot understand which app is displaying the PIN pad (i.e., if they are prompted by a trusted application in the TEE or from a malware app in the untrusted part of the OS that mimics the trusted one). This is due to the fact that the screen is shared among all apps installed on the smartwatch, independently from being hosted in the TEE or not. Furthermore, the input of the user's PIN code on smartwatches generates usability concerns, as a smartwatch screen is far smaller than a smartphone one. Thus, the design of an authentication method for smartwatches requires to deal with both security and usability concerns.

In this paper we put forward a novel idea for an authentication mechanism (*2GesturePIN*) able to enhance both the security and the usability of PIN input on smartwatches. The novelty of 2GesturePIN is to leverage the rotating bezel or crown of the smartwatch as a secure hardware (i.e., the bezel or crown is controlled by the TEE) to input four-PIN digits through solely two gestures. In this way, the user is not required to tap on small-size touch screens. This does not only improve the usability, but it also enhances the security of the regular PIN code against shoulder surfing, video recording, phishing and motion-based side channel attacks. Hence, 2GesturePIN could be used as an authentication mechanism for security critical NFC-based operations on smartwatches.

A typical usage scenario would be using 2GesturePIN to secure the access to an e-wallet that allows the user to open her house, car and office doors as well as making payment in store, paying parking and public transport tickets, just to cite

Fig. 1. 2GesturePIN user interface.

a few. In such a scenario, the user has to authenticate to the smartwatch and then she can perform any operation by simply tapping her smartwatch on NFC devices, without the need to carry multiple cards and keys or look for her smartphone in her pocket each time she wants to perform a transaction. At the same time, while a stolen card can be automatically used by the thief to impersonate the user (e.g., access a building or small-value payment), the stolen smartwatch cannot be used unless the thief has the owner's PIN as well.

The rest of this paper is structured as follows: In section II we introduce 2GesturePIN and we describe two implementations on two different smartwatch platforms; in section III we present related work; finally, section IV concludes this paper.

II. Introducing 2GesturePIN

The 2GesturePIN authentication framework is a user-centric security solution for smartwatch-based sensitive applications such as payment and access control. 2GesturePIN leverages smartwatches TEE security features and a novel authentication mechanism in order to enhance *i)* the usability of the authentication process and *ii)* the resiliency against a wide range of security threats.

Differently from a traditional PIN-input interface that uses a digital keypad, the 2GesturePIN UI consists of two concentric wheels with ten equally sized sectors as shown in Figure 1. The outer wheel contains numbers from 0 to 9 in a fixed order while the inner wheel is numbered randomly from 0 to 9 at each session and step. Furthermore, the inner wheel can rotate according to a TEE dedicated hardware movement, e.g., the movement of bezel or crown.

In order to authenticate, the user is required to rotate the inner wheel so that the first digit of his PIN matches the second one. In the same way, in the second step of the authentication process, the user rotates again a new randomly-generated inner wheel to match the third PIN digit with the fourth PIN digit. This implies that, thanks to the 2GesturePIN authentication method, the user is able to input his four-PIN digits with solely two simple gestures (i.e., rotating the wheel through the bezel) instead of typing it in small sized touch screen.

An example of authentication process using the 2GesturePIN Framework is shown in Figure 2 in the hypothesis that the user's PIN code is 7340.

In the first gesture, the user uses the bezel to rotate the first PIN digit (7) in the inner wheel in order to match the second PIN digit (3) of the outer wheel. Then, 2GesturePIN computes a second screen with a novel inner wheel with a random arrangement of the numbers. In the second gesture, the same process is repeated with the third and fourth PIN digits, the user rotates the the number 4 to match number 0.

For this description, we selected the smartwatch bezel as the dedicated hardware to exploit the trusted path to the TEE. Furthermore the user has installed on his smartwatch an application that requires the 2GesturePIN authentication mechanism. The application is composed by the Application UI (A) that resides in the Rich Execution Environment and contains the 2GesturePIN Library (AuthLib) and the corresponding trusted part (TA), placed in the TEE, with the 2GesturePIN Engine embedded inside (AuthEngine).

Once the user U taps on the application A icon, an authentication request is sent to the AuthLib and then dispatched to the *AuthEngine*.

At this point the *AuthEngine* computes the first random sequence of the wheel's numbers that will be used to retrieve the first pair of digits of the user's PIN. The sequence is then sent to AuthLib that renders the interface, as showed in Figure 2, and displayed it to U.

Once the authentication screen is displayed, the user rotates the bezel of the smartwatch in order to match the first pair of digits of his PIN and confirms its choice with the side button. After the confirmation, the Bezel driver, that resides in the TEE, gets the rotation degree of the bezel and sends this information to the *AuthEngine*.

The *AuthEngine* computes the second random sequence of the wheel's numbers necessary for the input of the second pair of PIN digits. After the second interaction with the user, the *AuthEngine* is able to check if the input provided matches the User's PIN.

At the end of the process, a notification - either success or failure - is sent to the TA and to the user through the smartwatch interface.

A. 2GesturePIN Implementation

One potential implementation of 2GesturePIN framework is through SierraTEE [7] which provides an open source implementation of TEE environment compatible with Global Platform standards and ARM TrustZone. SierraTEE supports multiple operating systems including any 64-bit platform utilizing ARM Cortex-A53 processors which are used by Samsung Gear S3 smartwatches. However, the actual presence of the TEE that is required to make the solution secure is meaningless from the usability point of view. Therefore, for usability test purpose, we implemented the 2GesturePIN on Tizen OS 2.3.2 as well as on Android Wear OS 2.0 with no dependencies to the TEE.

Fig. 2. 2GesturePIN authentication method.

Fig. 3. 2GesturePIN screenshots in Tizen emulator

We developed the applications using Tizen Studio 3.0 and Android Studio 3.1 respectively. The test equipment consisted of the Samsung Gear S3 Frontier LTE smartwatch from Samsung, which is based on Tizen 2.3.2 and fitted out with a hardware rotating bezel, a 360x360 pixels screen, a Dual-core 1.0 GHz, 768MB RAM and 4GB internal memory. For Android Wear, we used the LG Watch Style W270 smartwatch. Released at the beginning of 2017, it runs Android wear 2.0 and it features a Quad-Core 1.1 GHz Qualcomm Snapdragon Wear 2100 CPU, with 512 MB of RAM and 4 GB of internal memory and 360x360 pixels screen. Instead of the rotating bezel, this smartwatch is equipped with a rotating crown on the side.

As shown in the figure 3, the 2GesturePIN application has two modes, namely *training* and *test*, where the first mode allows participants to get familiar with the concept and the latter may be used for usability tests, in which we record the input time and the error rate on ten consecutive attempts. The application menu features also a settings button where we can update the PIN, along with other settings such as toggling the colors use, this option displays different colors on the wheel portions where each digit has always the same color.

III. Related work

Authentication on smartwatches is mainly used to set a lock screen to prevent unauthorized access to the device, and it is usually disabled by default. It is required, however, if the user wants to take advantage of mobile payment systems (e.g., Apple Pay, Google Pay or Samsung Pay) or controlling access to critical services and infrastructures (e.g., smart home Locker, smart cars locker, smart health services and infrastructures) in order to further tighten security around these systems. Although the known security and usability issues of the regular PIN and Pattern lock on smartwatches [2], [8], they are the predominant types of authentication mechanism today.

A. Behavioral-biometrics based authentication methods

This last motivated many researchers to take advantage of smartwatches rich sensing capabilities to design behavioral-biometrics based authentication methods as an alternative.

For instance, [9], [10] designed a motion-based authentication method able to authenticate a user by performing a gestures with a wrist worn device or smartwatch, after building the user's behavioural profile by collecting data from device sensors. A similar approach has been taken by SnapAuth [11], where the authentication is performed by a finger-snapping gesture. This system achieved 82.34% True Acceptance Rate (TAR) at 34.12% False Accept Rate (FAR) using one-class MLP as the classifier, on very limited training samples (i.e., 15). Johnston and Weiss [12] studied the feasibility of using smartwatches for gait-based biometrics. Their study shows that gait is not sufficient to be used as a sole means of identification of individuals; instead, it is seen as a potentially valuable component in a multimodal biometric system. Authors have also pointed out some limitations of their work: for instance, users data were collected the same day, thus not representing a real world scenario, and their preliminary works showed that results degrade significantly when data are collected on different days. A gait-based approach for continuous authentication has been investigated by [13]. Authors pointed out that gait recognition is highly efficient and recommended to

authenticate users in a transparent and continuous manner. Results are positive, however gait recognition and authentication were performed only in a controlled environment, thus results may differ in a real life scenario. None of these studies discuss the case where the user is not recognized by his walk or is not walking. A different approach has been taken by [14]: TapMeIn let the user authenticate by tapping a specific rhythm on the smartwatchs touchscreen. Results are significantly promising, with an accuracy of 98.7%, however tests were performed in a lab with a limited dataset, which may favour the classification process. All works presented above are related to a specific behavior of a human while performing some tasks, such as hand movement, gait, and rhythmic tapping, which may present some limitations [15]. For example, Multiple users may have the same hand waving patterns. Wearing an outfit, such as a trench coat or a footwear, may change a persons walking style and persons typing behavior changes considerably throughout a day with different states of mind such as excited, tired, etc. These limitations related to human behavior nature among others might be the main barriers to solely rely on a behavioral system.

B. Knowledge based authentication methods

Other researchers instead worked on increasing security around the current authentication methods: PIN and pattern lock. Research has focused on smartphones (e.g. [16]–[23]), ATM (e.g. [24]–[27]) and recently smartwatches, where the small screen size introduces usability concerns.

A novel PIN based authentication method is Personal Identification Chord (PIC) [28], where the user can enter ten different inputs using only four big on-screen buttons. The recall study shows that both PIN and PIC achieve high recall rates and input accuracy, however the usability study shows PIC as slightly slower and more error prone than PIN. Furthermore, PIC is not resilient to side-channel and shoulder-surfing attacks. In [29], authors introduced a two factor authentication method, called Draw-a-PIN. To authenticate, the user is required to draw his PIN digits sequentially on the touchscreen instead of typing it. Beside the correctness of the PIN, Draw-a-PIN uses the drawing behavior of the user as an additional security layer. While Draw-a-PIN provides some advantages with respect to shoulder-surfing resilience, the usability study of its implementation on a smartwatch [8] showed that it is not usable to unlock the smartwatch due to its high error rate and long authentication time (i.e., Overall Average Error rate 20.65%, Overall Average authentication time 7356 ms). Analogous method to TapMeIn [14] is Beat-PIN [30], where a PIN is represented by a sequence of beats recorded when the user taps on the smartwatch touchscreen (i.e, a beat is the time between the instance the user touches the screen and the instance the user lifts his finger from the screen). However, Beat-PIN does not use the user's typing behavior and thus it is less robust against shoulder surfing attack. Beat-PIN achieved an Equal Error Rate of 7.2% with an authentication time of 1.7 seconds. A sensors-based authentication is given by [31]. The variations of the lights values read by the ambient light sensor are used to build sequences representing particular User Interface (UI) events, such as single-click, double-click, 1-sec-hold, etc. These events are used to enter the PIN (e.g., a three events PIN could be single-click 1-sec-hold single-click). Besides its vulnerability to brute force and side channel attacks, this method is not usable because the input of solely a three events long PIN requires approximately between 9 and 10 seconds. In addition, using the ambient light sensor for the PIN input makes the input impossible in dark environment.

Similar to 2GesturePIN, VibraInput [32] and DialA [33] utilize two concentric wheels with ten equally sized sectors as a user interface for PIN authentication on smartphones. However, in contrast to 2GesturePIN, the outer wheel in DialA contains ten different letters and in VibraInput contains four repetitive letters while, each letter represents a vibration pattern. In addition to the four-PIN digits, the user has to remember four vibrations pattern and their corresponding letters. When the user touch the screen, the vibration starts and the user has to remember the letter that correspond to this vibration pattern in order to use it as an indicator to input the PIN digit. The vibration stops as soon as the user left his finger. Since the outer wheel contains multiple occurrences of this indicator, another round is required to identify the PIN digit. Thus, beside the overhead of memorizing an additional secret, VibraInput requires eight gestures to input four PIN digits. In DialA [33], the user has to use the letter that he heard through an earphones as an indicator to input the four-PIN digits. The rotation and commitment are conducted via another small scroll wheel at the bottom of the smartphone screen in order to prevent a direct input. However this method is not suitable for smartwatches for two reasons. First because the user is required to wear an earphones and connect it to the smartwatch through bluetooth (i.e., if it is not connected) which is impractical and take a long time. Second, because using another small wheel is not suitable for small-size screen such as smartwatches and rotating the wheel directly makes the user susceptible to side channel, shoulder surfing and video recording attacks. In addition, unlike 2GesturePIN, DialA requires four gestures.

IV. Conclusion

Nowadays, wearable transactions are becoming very popular due to the facilities provided by the NFC technology. However, the authentication step is often considered as the weakest link in the security of these transactions due to the increase of security threats targeting it. This paper introduced a novel PIN-based authentication method for smartwatches through two bezel (or crown) rotation gestures. Our preliminary study showed that the proposed scheme is resilient against brute force, phishing attacks, side channel, shoulder surfing and video-recording attacks. In future work, in order to have a complete assessment of 2GesturePIN, we intend to provide a formal threat model, to carry on a complete security analysis, to provide a complete description of all the software components needed to have a complete and secure implementation framework and, finally, an analysis of the requirements of

our framework for the Trusted Execution Environment. We aim at building a robust solution which allow to be resilient against any malware that could run on the smartwatch, as we argue that it is rather hard today to recognize and isolate malware activities on a smartwatch, also through non-standard approaches [34], [35]. In general, we argue that assuming that the smartwatch is untrusted, apart from the TEE part, is a reasonable worst case scenario hypothesis for validating the 2GesturePIN approach.

References

[1] A. Technologies, "Arm security technology building a secure system using trustzone technology," Tech. Rep., 2005. [Online]. Available: http://infocenter.arm.com/help/topic/com.arm.doc.prd29-genc-009492c/$PRD29-GENC-009492C_trustzone_security_whitepaper.pdf$.

[2] C. X. Lu, B. Du, H. Wen, S. Wang, A. Markham, I. Martinovic, Y. Shen, and A. Trigoni, "Snoopy: Sniffing your smartwatch passwords via deep sequence learning," *IMWUT*, vol. 1, 152:1–152:29, 2017.

[3] A. Brandon and M. Trimarchi, "Trusted display and input using screen overlays," in *2017 International Conference on ReConFigurable Computing and FPGAs (ReConFig)*, Dec. 2017, pp. 1–6. DOI: 10.1109/RECONFIG.2017.8279826.

[4] H. Khan, U. Hengartner, and D. Vogel, "Evaluating attack and defense strategies for smartphone pin shoulder surfing," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, ser. CHI '18, Montreal QC, Canada: ACM, 2018, 164:1–164:10, ISBN: 978-1-4503-5620-6. DOI: 10.1145/3173574.3173738. [Online]. Available: http://doi.acm.org/10.1145/3173574.3173738.

[5] G. Ye, Z. Tang, D. Fang, X. Chen, W. Wolff, A. J. Aviv, and Z. Wang, "A video-based attack for android pattern lock," *ACM Trans. Priv. Secur.*, vol. 21, no. 4, 19:1–19:31, Jul. 2018, ISSN: 2471-2566. DOI: 10.1145/3230740. [Online]. Available: http://doi.acm.org/10.1145/3230740.

[6] E. Owusu, J. Han, S. Das, A. Perrig, and J. Zhang, "Accessory: Password inference using accelerometers on smartphones," in *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications*, ser. HotMobile '12, San Diego, California: ACM, 2012, 9:1–9:6, ISBN: 978-1-4503-1207-3. DOI: 10.1145/2162081.2162095. [Online]. Available: http://doi.acm.org/10.1145/2162081.2162095.

[7] Sierraware, "Sierratee trusted execution environment," [Online]. Available: https://www.sierraware.com/open-source-ARM-TrustZone.html.

[8] "Smartwatches locking methods: A comparative study," in *Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017)*, Santa Clara, CA: USENIX Association, 2017. [Online]. Available: https://www.usenix.org/conference/soups2017/workshop-program/way2017/nguyen.

[9] J. Yang, Y. Li, and M. Xie, "Motionauth: Motion-based authentication for wrist worn smart devices," in *2015 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, Mar. 2015, pp. 550–555. DOI: 10.1109/PERCOMW.2015.7134097.

[10] A. Lewis, Y. Li, and M. Xie, "Real time motion-based authentication for smartwatch," in *2016 IEEE Conference on Communications and Network Security (CNS)*, Oct. 2016, pp. 380–381. DOI: 10.1109/CNS.2016.7860521.

[11] A. Buriro, B. Crispo, M. Eskandri, S. Gupta, A. Mahboob, and R. Van Acker, "Snapauth: A gesture-based unobtrusive smartwatch user authentication scheme," in *Emerging Technologies for Authorization and Authentication*, A. Saracino and P. Mori, Eds., Cham: Springer International Publishing, 2018, pp. 30–37, ISBN: 978-3-030-04372-8.

[12] A. H. Johnston and G. M. Weiss, "Smartwatch-based biometric gait recognition," in *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, Sep. 2015, pp. 1–6. DOI: 10.1109/BTAS.2015.7358794.

[13] N. Al-Naffakh, N. Clarke, F. Li, and P. Haskell-Dowland, "Unobtrusive gait recognition using smartwatches," in *2017 International Conference of the Biometrics Special Interest Group (BIOSIG)*, Sep. 2017, pp. 1–5. DOI: 10.23919/BIOSIG.2017.8053523.

[14] T. Nguyen and N. Memon, "Tap-based user authentication for smartwatches," *Computers & Security*, vol. 78, pp. 174–186, 2018, ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2018.07.001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404818303778.

[15] M. Ehatisham-ul-Haq, M. A. Azam, J. Loo, K. Shuang, S. Islam, U. Naeem, and Y. Amin, "Authentication of smartphone users based on activity recognition and mobile sensing," *Sensors*, vol. 17, no. 9, 2017, ISSN: 1424-8220. DOI: 10.3390/s17092043. [Online]. Available: http://www.mdpi.com/1424-8220/17/9/2043.

[16] M. Guerar, M. Migliardi, A. Merlo, M. Benmohammed, F. Palmieri, and A. Castiglione, "Using screen brightness to improve security in mobile social network access," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 621–632, Jul. 2018, ISSN: 1545-5971. DOI: 10.1109/TDSC.2016.2601603.

[17] E. von Zezschwitz, A. De Luca, B. Brunkow, and H. Hussmann, "Swipin: Fast and secure pin-entry on smartphones," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ser. CHI '15, Seoul, Republic of Korea: ACM, 2015, pp. 1403–1406, ISBN: 978-1-4503-3145-6. DOI: 10.1145/2702123.2702212. [Online]. Available: http://doi.acm.org/10.1145/2702123.2702212.

[18] M. Guerar, M. Migliardi, A. Merlo, M. Benmohammed, and B. Messabih, "A completely automatic public phys-

ical test to tell computers and humans apart: A way to enhance authentication schemes in mobile devices," in *2015 International Conference on High Performance Computing Simulation (HPCS)*, Jul. 2015, pp. 203–210. DOI: 10.1109/HPCSim.2015.7237041.

[19] M. Guerar, A. Merlo, and M. Migliardi, "Completely automated public physical test to tell computers and humans apart: A usability study on mobile devices," *Future Generation Computer Systems*, vol. 82, pp. 617–630, 2018, ISSN: 0167-739X. DOI: https://doi.org/10.1016/j.future.2017.03.012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X17303709.

[20] A. Bianchi, I. Oakley, V. Kostakos, and D. S. Kwon, "The phone lock: Audio and haptic shoulder-surfing resistant pin entry methods for mobile devices," in *Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction*, ser. TEI '11, Funchal, Portugal: ACM, 2011, pp. 197–200, ISBN: 978-1-4503-0478-8. DOI: 10.1145/1935701.1935740. [Online]. Available: http://doi.acm.org/10.1145/1935701.1935740.

[21] T. Kwon and S. Na, "Tinylock: Affordable defense against smudge attacks on smartphone pattern lock systems," *Computers & Security*, vol. 42, pp. 137–150, 2014, ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2013.12.001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404813001697.

[22] M. Guerar, A. Merlo, and M. Migliardi, "Clickpattern: A pattern lock system resilient to smudge and side-channel attacks," *JoWUA*, vol. 8, no. 2, pp. 64–78, 2017. [Online]. Available: http://isyou.info/jowua/papers/jowua-v8n2-4.pdf.

[23] M. Guerar, A. Merlo, M. Migliardi, and F. Palmieri, "Invisible cappcha: A usable mechanism to distinguish between malware and humans on the mobile iot," *Computers Security*, vol. 78, pp. 255–266, 2018, ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2018.06.007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404818307557.

[24] M. Guerar, M. Benmohammed, and V. Alimi, "Color wheel pin: Usable and resilient ATM authentication," *J. High Speed Networks*, vol. 22, no. 3, pp. 231–240, 2016. DOI: 10.3233/JHS-160545. [Online]. Available: https://doi.org/10.3233/JHS-160545.

[25] A. D. Luca, E. von Zezschwitz, and H. Hußmann, "Vibrapass: Secure authentication based on shared lies," in *CHI*, ACM, 2009, pp. 913–916.

[26] A. De Luca, K. Hertzschuch, and H. Hussmann, "Colorpin: Securing pin entry through indirect input," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '10, Atlanta, Georgia, USA: ACM, 2010, pp. 1103–1106, ISBN: 978-1-60558-929-9. DOI: 10.1145/1753326.1753490. [Online]. Available: http://doi.acm.org/10.1145/1753326.1753490.

[27] D. Nyang, A. Mohaisen, and J. Kang, "Keylogging-resistant visual authentication protocols," *IEEE Transactions on Mobile Computing*, vol. 13, no. 11, pp. 2566–2579, Nov. 2014, ISSN: 1536-1233. DOI: 10.1109/TMC.2014.2307331.

[28] I. Oakley, J. H. Huh, J. Cho, G. Cho, R. Islam, and H. Kim, "The personal identification chord: A four buttonauthentication system for smartwatches," in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, ser. ASIACCS '18, Incheon, Republic of Korea: ACM, 2018, pp. 75–87, ISBN: 978-1-4503-5576-6. DOI: 10.1145/3196494.3196555. [Online]. Available: http://doi.acm.org/10.1145/3196494.3196555.

[29] T. V. Nguyen, N. Sae-Bae, and N. Memon, "Draw-a-pin: Authentication using finger-drawn pin on touch devices," *Computers & Security*, vol. 66, pp. 115–128, 2017, ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2017.01.008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404817300123.

[30] B. Hutchins, A. Reddy, W. Jin, M. Zhou, M. Li, and L. Yang, "Beat-pin: A user authentication mechanism for wearable devices through secret beats," in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, ser. ASIACCS '18, Incheon, Republic of Korea: ACM, 2018, pp. 101–115, ISBN: 978-1-4503-5576-6. DOI: 10.1145/3196494.3196543. [Online]. Available: http://doi.acm.org/10.1145/3196494.3196543.

[31] H. Yoon, S. Park, and K. Lee, "Exploiting ambient light sensor for authentication on wearable devices," in *2015 Fourth International Conference on Cyber Security, Cyber Warfare, and Digital Forensic (CyberSec)*, Oct. 2015, pp. 95–100. DOI: 10.1109/CyberSec.2015.27.

[32] T. Kuribara, B. Shizuki, and J. Tanaka, "Vibrainput: Two-step pin entry system based on vibration and visual information," in *CHI '14 Extended Abstracts on Human Factors in Computing Systems*, ser. CHI EA '14, Toronto, Ontario, Canada: ACM, 2014, pp. 2473–2478, ISBN: 978-1-4503-2474-8. DOI: 10.1145/2559206.2581187. [Online]. Available: http://doi.acm.org/10.1145/2559206.2581187.

[33] M.-K. Lee, H. Nam, and D. K. Kim, "Secure bimodal pin-entry method using audio signals," *Computers Security*, vol. 56, pp. 140–150, 2016, ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2015.06.006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404815000929.

[34] M. Migliardi and A. Merlo, "Modeling the energy consumption of distributed ids: A step towards green security," in *2011 Proceedings of the 34th International Convention MIPRO*, May 2011, pp. 1452–1457.

[35] A. Merlo, M. Migliardi, and P. Fontanelli, “Measuring and estimating power consumption in android to support energy-based intrusion detection,” *Journal of Computer Security*, vol. 23, no. 5, pp. 611–637, 2015. DOI: 10 . 3233 / JCS - 150530. [Online]. Available: http : / / www . scopus . com / inward / record . url ? eid = 2 - s2 . 0 - 84948156627 & partnerID = 40 & md5 = 406073c0920c25e7c2c009e06f9c246a.