SwipeVLock

A Supervised Unlocking Mechanism Based on Swipe Behavior on Smartphones

Li, Wenjuan; Tan, Jiao; Meng, Weizhi; Wang, Yu; Li, Jing

Published in:
Proceedings of 2nd International Conference on Machine Learning for Cyber Security

Link to article, DOI:
10.1007/978-3-030-30619-9_11

Publication date:
2019

Document Version
Peer reviewed version

SwipeVLock: A Supervised Unlocking Mechanism Based on Swipe Behavior on Smartphones

Wenjuan Li[1,2], Jiao Tan[3], Weizhi Meng[1,4(✉)], Yu Wang[1], and Jing Li[1]

[1] School of Computer Science, Guangzhou University, Guangzhou, China
weme@dtu.dk
[2] Department of Computer Science, City University of Hong Kong, Kowloon, China
[3] KOTO Research Center, Macao, China
[4] Department of Applied Mathematics and Computer Science, Technical University of Denmark, Lyngby, Denmark

Abstract. Smartphones have become a necessity in people's daily lives, and changed the way of communication at any time and place. Nowadays, mobile devices especially smartphones have to store and process a large amount of sensitive information, i.e., from personal to financial and professional data. For this reason, there is an increasing need to protect the devices from unauthorized access. In comparison with the traditional textual password, behavioral authentication can verify current users in a continuous way, which can complement the existing authentication mechanisms. With the advanced capability provided by current smartphones, users can perform various touch actions to interact with their devices. In this work, we focus on swipe behavior and aim to design a machine learning-based unlock scheme called SwipeVLock, which verifies users based on their way of swiping the phone screen with a background image. In the evaluation, we measure several typical supervised learning algorithms and conduct a user study with 30 participants. Our experimental results indicate that participants could perform well with SwipeVLock, i.e., with a success rate of 98% in the best case.

Keywords: User authentication · Behavioral biometric · Swipe behavior · Smartphone security · Touch action

1 Introduction

With the revolution of information technology, mobile devices like smartphones have become prevalent in people's lives. More users are willing to store private information on their devices and use them to process some sensitive information for mobility and convenience [27,50]. However, this also makes smartphones a major target by cyber-criminals [31]. If attackers get the phone and unlock it successfully, then they can easily steal all sensitive data. Thus, there is a

X. Chen et al. (Eds.): ML4CS 2019, LNCS 11806, pp. 140–153, 2019.
https://doi.org/10.1007/978-3-030-30619-9_11

demanding requirement for implementing user authentication mechanisms to prevent unauthorized access.

Up to now, the most widely adopted authentication approach is still based on textual passwords. For example, iPhones use PIN code to protect the devices, but it may suffer many invasions, e.g., recording attacks [31]. In real-world applications, users have multiple accounts and may choose easy-to-remember passwords due to the multiple password inference [28] and limitation of long term memory [49]. Some research studies like [3,48] revealed that this situation may become even worse under existing state-of-the-art attacks. For example, the report from SplashData showed that the most frequently used password in 2018 is "123456" [39].

As an alternative, graphical passwords (GP) were developed to enhance the authentication process, since many studies like [30,36] identified that people could remember images better than string passwords. There are many GP schemes in the literature. For instance, Jermyn *et al.* [14] introduced DAS (draw-a-secret) that requires users to draw their passwords on a 2D grid. Wiedenbeck *et al.* [47] developed *PassPoints* that allows creating users' credentials by clicking on some locations on an image. In practice, GP schemes are not widely adopted by mobile devices, but there exists a typical application called *Android unlock patterns*, which requires users inputting correct patterns to unlock their phones in a grid size of 3×3 points [2,7]. For authentication, users have to recall the pattern registered during the enrollment.

However, Android unlock patterns may be vulnerable to many attacks in real-world usage, as users can only choose a pattern with 4 dots at least and 9 dots at most. This makes Brute-force attack feasible because the total number of possible patterns is only 389,112 [1]. In addition, it also suffers recording attacks [31] and charging attacks [25,26] (i.e., the phone screen can be captured by attackers). As a result, there is a great demand to enhance the security of such unlocking mechanism.

Contributions. Many existing research studies have shown that combining behavioral biometric could provide an additional security layer to safeguard the Android unlock patterns [7,17,52]. For example, De Luca et al. [7] showed how to combine behavioral biometric with unlock patterns using dynamic time warping (DTW). Motivated by this, in this work, we advocate the merit of enhancing authentication with behavioral biometric, and develop SwipeVLock, a swipe behavior-based unlock mechanism on smartphones. In our scheme, users can choose a background image and a location on the image to swipe their finger. The contributions of this work can be summarized as follows.

- We design SwipeVLock, a phone unlocking scheme that verifies users based on how they swipe the touchscreen. For enrollment, users have to choose one background image and one location, and then register their swipe behavior. This mechanism is transparent without additional hardware on smartphones. We also test several typical supervised learning algorithms for authentication.
- In the user study, we involve a total of 30 common phone users to evaluate the performance of SwipeVLock. Based on the collected data and users' feedback,

it is found that our scheme can provide good usability in practice. SwipeVLock can be considered as one alternative to complement existing solutions.

Road Map. The rest of this paper is structured as follows. Section 2 introduces related authentication schemes based on either graphical passwords or touch behavioral biometric. Section 3 describes our scheme of SwipeVLock in detail. In Sect. 4, we conduct a user study with 30 participants and analyze the collected data. We discuss some open challenges and conclude our work in Sects. 5 and 6.

2 Related Work

This section introduces related studies regarding graphical passwords schemes and touch behavioral authentication.

2.1 Authentication Based on Graphical Password

Graphical passwords have been researched over decades. There are three major types for a traditional GP scheme [4,29,42]: recognition-based scheme, pure recall-based scheme and cued recall-based scheme.

- *Recognition-based scheme.* This kind of scheme (e.g., [6,32]) needs users to remember and recognize several images. Taking *PassFaces* [32] as a typical example, it requires users to figure out human faces for user authentication.
- *Pure recall-based scheme.* This type of scheme requires users to generate a pattern on an image. For example, Jermyn *et al.* [14] introduces *DAS* ('draw-a-secret'), in which users have to create their passwords on a grid. Android unlock pattern (AUP) mechanism belongs to this type, asking users to swipe their finger to input a correct pattern and unlock the device. It is indeed a modified version of Pass-Go [44], in order to fit a small touchscreen. AUP has some rules, i.e., it defines a valid pattern with 4 dots at least and 9 dots at most, within a grid of 3×3 points on smartphones.
- *Cued recall-based scheme.* Such schemes require users to create a pattern on an image or more images. Taking a typical system of *PassPoints* [47] as an example, it needs users to remember five points on one image in an order. Then Chiasson *et al.* [5] introduced Persuasive Cued Click-Points (PCCP), in which users have to pick a point on a sequence of background images.

In addition to the above major schemes, existing GP schemes are more integrated. For example, with the aim of enhancing the password space, world map has been proposed as the background image, in which users can choose a location worldwide [11,38]. Based on this idea, Sun *et al.* [43] designed *PassMap* that requires users to choose two locations (in an order) on a world map. Then Thorpe *et al.* [45] introduced *GeoPass* that only requires users to select one location. The previous study showed that there is no significant difference between the selection of one or two locations [29]. Meng [22] designed *RouteMap*, a map-based scheme that demands users to create a route on a world map.

Similar to textual passwords, graphical passwords may also suffer the issue of multiple password interference. Meng *et al.* [28] investigated this issue with 60 participants between textual passwords and map-based passwords under six account scenarios. They found that participants in the map-based graphical password scheme could perform better than the textual password scheme in both short-term (one-hour session) and long term (after two weeks) password memorability tests.

To further enhance the performance of graphical passwords, there is a balance should be made between security and usability. A set of hybrid GP schemes were also developed in the literature, like click-draw based GP scheme [17]. Some relevant GP studies could be referred but not limited to [8,9,13,16–20,23,24,27,51].

2.2 Touch Behavioral Authentication

With the advent of touchscreen, touch dynamics has become popular on smartphones. Fen *et al.* [10] developed a finger gesture-based authentication system on touchscreen devices, reaching a FAR of 4.66% and a FRR of 0.13% based on a random forest classifier. Meng *et al.* [17] validated the feasibility of touch behavioral authentication on smartphones, where they designed scheme with 21 features and achieved an average error rate of around 3% based on a combined classifier of PSO-RBFN. Frank *et al.* [12] developed *Touchalytics*, a touch behavioral authentication scheme with 30 features, and reached a median equal error rate of around 4% (one week after the enrollment phase).

Up to now, more touch behavioral authentication schemes have been proposed [21]. Zheng *et al.* [53] researched users' tapping behaviors on a passcode-enabled smartphone, and achieved an averaged equal error rate of nearly 3.65% by using a one-class algorithm. Smith-Creasey and Rajarajan [37] achieved an equal error rate of 3.77% by means of a stacked classifier approach. Sharma and Enbody [41] studied how users interact with the application interface, and achieved a mean equal error rate of 7% for user authentication based on the SVM-based ensemble classifier. Shahzad *et al.* [40] researched users' particular behavior and designed an authentication scheme based on how users input a gesture or a signature, such as velocity, device acceleration, and stroke time.

3 Design of SwipeVLock

The purpose of our proposed SwipeVLock is to complement existing unlocking mechanisms on smartphones, through involving touch behavioral authentication. Figure 1 shows the basic design of SwipeVLock with three major steps.

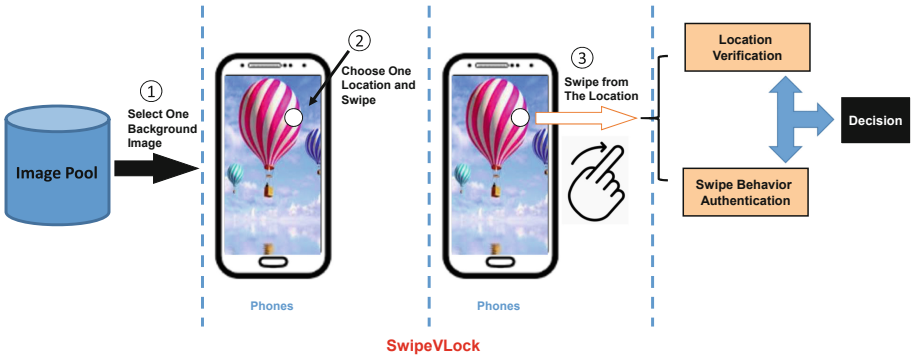

Fig. 1. SwipeVLock: (1) Step1: select one background image from a pool; (2) Step2: choose one location on the background image; and (3) Step3: swipe from the selected location to unlock the phone.

SwipeVLock Enrollment. Users have to select one background image from an image pool, with different themes such as fruits, cartoon characters, sport, landscape, food, buildings, transportation, people, etc. Then, users can choose one location as the starting point and then swipe the screen from this selected location.

SwipeVLock Verification. For authentication, users have to select the same background image from the pool, and swipe the screen from the same location on the image. The authentication process can be regarded to be successful, if and only if both image location and swipe behavior are verified by our scheme.

SwipeVLock Framework. Figure 2 depicts how to realize SwipeVLock. In this work, our scheme employs a supervised learning-based framework to help model users' touch behavior. When users swipe the screen, SwipeVLock will extract the touch features from swipe behavior and train the classifier. The classifier mainly generates a normal profile based on the swipe behavior, and compares it with the current swipe features. A decision will be output in the end.

On the other hand, SwipeVLock can compare the image location with the stored location in the database. If there is a match, then it is considered to be successful. In particular, we set the error tolerance to a 21×21 pixel box around the selected location. This selection is based on the previous work like [29,45]. For example, *GeoPass* [45] proved that an error tolerance of 21×21 pixel is usable in practice.

Swipe Features. In this work, based on the previous studies [7,12,24], we consider some common and typical touch features that can be used to model swipe behavior: the coordinates of location (XY), touch pressure, touch size, touch time, and touch speed.

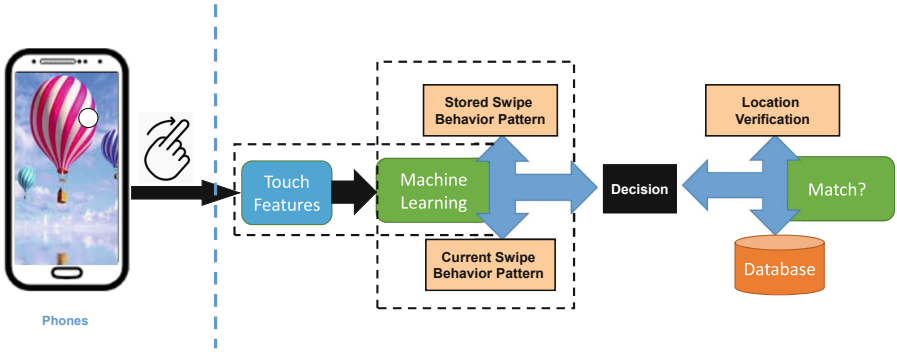

Fig. 2. Detailed authentication processes for SwipeVlock.

- *Coordinates of location.* Our scheme records the location coordinates on the selected image. Intuitively, users may have their own selection preference, making the location different from others.
- *Touch pressure.* With the increasing capability of smartphones, current screen sensors are able to identify the values of touch pressure, which can be used to model users' touch behavior.
- *Touch duration.* This feature can be computed by measuring the time difference between touch press-down and touch press-up. It is a common feature that can be used to distinguish different users, i.e., some users may press longer while some may press shorter.
- *Touch speed.* Intuitively, swipe behavior can be treated as a swift touch movement. Based on [24], suppose a swipe action starts from (x1, y1) and ends at (x2, y2), if we know relevant time of occurrence *T1* and *T2*, then we can calculate the touch speed according to Eq. (1).

$$Touch\ Speed = \frac{\sqrt{(x2 - x1)^2 + (y2 - y1)^2}}{T2 - T1} \tag{1}$$

4 User Study

To investigate the performance of our scheme, we perform a user study with 30 participants who are regular Android phone users. The detailed information is shown in Table 1. In particular, we have 17 males and 13 females who aged from 18 to 45. Most of them are students in addition to business people, university staff and faculty members. A $20 gift voucher was provided to each participant.

Supervised Learning. As mentioned in Fig. 2, SwipeVLock uses supervised learning algorithms to help verify users. In this work, we consider the following classifiers as a study: Decision tree (J48), Naive Bayes, SVM and Back Propagation Neural Network (BPNN). These are the typical and popular classifiers in the literature.

Table 1. Participants information in the user study.

Information	Male	Female	Occupation	Male	Female
Age <25	10	7	Students	13	10
Age 25–35	4	4	University Faculty&Staff	2	2
Age 35–45	3	2	Business People	2	1

- J48 is a decision tree classifier [33], which can label data based on the pre-trained tree-like structure.
- Naive Bayes is kind of supervised learning algorithms based on Bayes theorem by assuming conditional independence between every pair of features given the value of the class variable [34].
- BPNN is a kind of neural network classifier [35], which uses a differentiable transfer function at each network node and then uses error back-propagation process to modify the internal network weights after each training round.
- Support Vector Machine (SVM) [15] is a linear model for both classification or regression challenges, by generating a line or a hyperplane that separates the data into classes.

To avoid any bias during classifier implementation, we adopted WEKA platform, which is an open-source machine learning collection in Java [46]. We used the default settings for all classifiers in the study. Below are two metrics used to evaluate the performance of our scheme.

- False Acceptance Rate (FAR): indicates the percent of how many intruders are classified as normal users.
- False Rejection Rate (FRR): indicates the percent of how many legitimate users are classified as intruders.

Study Steps. In the study, we first introduced our objectives to all participants and demonstrated what kind of data would be collected. Each participant could get one Android phone (Samsung Galaxy Note) and before the experiment, each of them has three trials to get familiar with the scheme. Then we randomly divided participants into two groups. In particular, Group-A was asked to perform the experiment in our lab, while the participants in Group-B could set their SwipeVLock in the lab and keep using the phone outside. Below are the detailed study steps.

- *Group1.* Participants in this group were required to complete the experiment in the lab.
  - Step 1. Creation phase: participants have to create their credentials according to SwipeVLock' steps.
  - Step 2. Confirmation phase: participants should confirm the password by verifying both the image location and swipe behavior for 10 times (used for classifier selection). Participants could modified their credentials if they fail or want to change it.

  - Step 3. Distributed memory: participants were provided one paper-based finding tasks to distract them for 15 min.
  - Step 4. Login phase: participants should swipe to unlock the phone for 10 trials. The system recorded all the data for analysis.
  - Step 5. Feedback form: participants should respond to several questions in a *feedback form* regarding our scheme usage.
  - Step 6. Retention. After three days, participants were asked to return and unlock the phone for 10 times in our lab.
  - Step 7. Participants have to finish another *feedback from* regarding our scheme usage.
- *Group2.* Participants in this group could create their SwipeVLock credentials in the lab, and then keep using the phone outside the lab.
  - Step 1. Creation phase: participants have to create their credentials according to SwipeVLock' steps.
  - Step 2. Confirmation phase: participants should confirm the password by verifying both the image location and swipe behavior for 10 times (used for classifier selection). Participants could modified their credentials if they fail or want to change it.
  - Step 3. Distributed memory: participants were provided one paper-based finding tasks to distract them for 15 min.
  - Step 4. Login phase: participants should swipe to unlock the phone for 10 trials. The system recorded all the data for analysis.
  - Step 5. Feedback form: participants should respond to several questions in a *feedback form* regarding our scheme.
  - Step 6. Retention. Participants could keep the phone and try to unlock the phone at last once each day. After three days, participants were asked to return and unlock the phone for 10 times in our lab.
  - Step 7. Participants have to finish another *feedback from* regarding our scheme.

Study Results. In the confirmation phase, we could collect 150 trials in the login phase for each Group1 and Group2. We used 60% of them as training data and the rest as testing data (with a cross-validation mode). The performance of different classifiers is depicted in Table 2. It is found that SVM could achieve a smaller error rate than other classifiers, i.e., it could reach an AER of 4.1% and 4.45% in Group1 and Group2, respectively. In contrast, BPNN could reach an AER of around 7%, while J48 & NBayes may cause an AER over 10%.

In this case, we used SVM as the classifier in SwipeVLock. Table 3 shows the successful unlock trials for login phase and retention phase in Group1 and Group2.

- *Login phase.* It is observed that participants in both groups could perform well with a success rate of 97.3% (Group1) and 95.3% (Group2), respectively. The errors were mainly caused by behavioral deviation, i.e., some participants may perform a swipe too fast.

Table 2. The performance of different classifiers under different groups.

Group1	J48	NBayes	SVM	BPNN	*Group2*	J48	NBayes	SVM	BPNN
FAR (%)	9.7	12.4	3.7	6.8	FAR (%)	10.6	11.5	4.1	6.8
FRR (%)	10.3	10.3	4.5	7.2	FRR (%)	11.3	12.2	4.8	7.6
AER (%)	10.0	11.35	4.1	7.0	AER (%)	10.95	11.85	4.45	7.2

Table 3. Success rate in the login and retention phase for Group1 and Group2.

Login	Group1	Group2
Success rate	146/150 (97.3%)	143/150 (95.3%)
Retention	Group1	Group2
Success rate	132/150 (88%)	147/150 (98%)

– *Retention phase.* After three days, it is found that participants in Group2 performed much better than those in Group1. This is because participants in Group2 could keep the phone and practice the unlocking behavior. Some participants reported that they might unlock the phone 16 times a day, making their swipe behavior more stable.

It is interesting to notice there are fewer errors caused by location selection, indicating that the error tolerance is suitable in practical usage. Further, our results validate that more practice can make the touch behavior more stable, which is in-line with the observations in [24]. For the retention phase in Group2, participants achieved a success rate of 98%, which is promising in real-world applications.

User Feedback. During the study, we gave two feedback forms to each participant regarding the scheme usage. Ten-point Likert scales were used in each feedback question, where 1-score indicates strong disagreement and 10-score indicates strong agreement. Several key questions and scores are summarized in Table 4.

– *Group1.* Most participants were satisfied with the usage of SwipeVLock, resulting in a score of over 8.5 on average for each question. We informally interviewed 10 of them, and they believed this is an easy-to-use unlock mechanism.
– *Group2.* The participants in Group2 provided a higher score than Group1, i.e., 9.1 vs. 8.7 for the third question. The reason may be due to that the participants in this group could keep the phone and try it for three days. We also informally interviewed 12 of them, and found that they had fun of using this mechanism. Most of them have an interest to use it in their own phones.

Table 4. Major questions and average scores received from the user study.

Questions (Group1)	Average scores
1. I could easily create a credential under SwipeVLock	8.8
2. The time consumption for SwipeVLock creation is acceptable	8.5
3. I could easily login to the system	8.7
Questions (Group2)	Average scores
1. I could easily create a credential under SwipeVLock	9.0
2. The time consumption for SwipeVLock creation is acceptable	8.7
3. I could easily login to the system	9.1

Though users' feedback is a subjective way of evaluating the scheme performance, it still provides valuable comments on our scheme. For instance, in the study, we received many positive answers, which can support and motivate the development of SwipeVLock, i.e., some participants are willing to use our mechanism on their own phones. We consider that our scheme could become a promising alternative to complement existing unlock mechanisms on smartphones.

5 Discussion

In the user study, we obtain promising results on the usage of our scheme. However, our work is still an early study to explore the performance of SwipeVLock, there are many challenges and limitations.

- *Time consumption.* In this work, we did not investigate the time consumption, as it normally takes less than 10 s. Most participants also satisfied with the login time in our feedback forms. In our future work, we plan to perform a larger study to explore this issue.
- *Image selection.* The first step of SwipeVLock is to select one background image from a pool (i.e., with 10 images). Intuitively, users have their own preference and are likely to choose a different image. However, with more users, it is unclear whether there would be a bias. This is an interesting topic in our future work.
- *Location selection.* The second step of SwipeVLock is to choose a location on the selected image. Similar to the image selection, it is also an interesting topic to investigate whether there is a selection bias, and explore which part of image is most likely to be selected.
- *More participants.* In this work, we mainly involved 30 participants in the study. In our future work, we plan to recruit more participants with diverse background to validate our results. In addition, it is also an interesting topic to investigate the difference between right handed and left handed participants, and check the observations.

- *Advanced attacks.* Our focus in this work is to investigate the performance of SwipeVLock, we did not consider some adversarial scenarios, where an attacker may get the phone and try to unlock it. This is an important topic in our future work, i.e., exploring the effect of recording attacks and mimic attacks.
- *Multi-touch behavior.* At this stage, our scheme only considers a swipe action with single finger, while it is an interesting topic to investigate the performance by using two fingers.
- *Phone type.* In this work, we mainly used one type of Android phone in the user study, while it could be an interesting topic to explore whether phone models may affect the scheme performance. This is also an open challenge for existing authentication schemes.
- *Machine leaning.* Supervised learning algorithms are widely adopted when designing a user authentication scheme [21]. In this work, we considered some common and popular machine learning schemes to model users' behavior. Our future work plans to involve more diverse learning algorithms, e.g., ensemble algorithms, and to investigate the effect of feature distance approaches.
- *Comparison with other schemes.* Our study focuses on evaluating the performance of SwipeVLock itself, while we plan to consider a comparison with similar schemes in future. For example, we can include some existing graphical password schemes, behavioral schemes or hybrid schemes. This is an open challenge in this area, as there lacks a unified platform for comparison.

6 Conclusion

Unlock mechanisms like Android unlock patterns are an important security tool to protect smartphones from unauthorized access, but attackers can still compromise the phone via various attacks like shoulder surfing, recording attacks and charging attacks. As a result, there is an increasing need to enhance the security of unlock mechanisms. In this work, motivated by this issue, we develop SwipeVLock, a swipe behavior-based unlock scheme with a supervised framework on smartphones, which requires users to choose one background image and a location to perform a swipe action. A successful trial should have both successful location selection and swipe verification. In our user study, we involved a total of 30 participants and investigated their performance like success rate. Our results demonstrate that participants could reach a success rate of 98% in the best scenario. Most participants also provide positive feedback on the practical usage of SwipeVLock.

Acknowledgments. We would like to thank the participants for their hard work in the user study. This work was partially supported by National Natural Science Foundation of China (No. 61802077).

References

1. Aviv, A.J., Gibson, K., Mossop, E., Blaze, M., Smith, J.M.: Smudge attacks on smartphone touch screens. In: Proceedings of the 4th USENIX Conference on Offensive Technologies, pp. 1–7. USENIX Association (2010)
2. Berkeley Churchill, Unlock Pattern Generator (2013). https://www.berkeleychurchill.com/software/android-pwgen/pwgen.php
3. Bonneau, J.: The science of guessing: analyzing an anonymized corpus of 70 million passwords. In: Proceedings of the 2012 IEEE Symposium on Security and Privacy, pp. 538–552 (2012)
4. Chiasson, S., Biddle, R., van Oorschot, P.C.: A second look at the usability of click-based graphical passwords. In: Proceedings of the 3rd Symposium on Usable Privacy and Security (SOUPS), pp. 1–12. ACM, New York (2007)
5. Chiasson, S., Stobert, E., Forget, A., Biddle, R.: Persuasive cued click-points: design, implementation, and evaluation of a knowledge-based authentication mechanism. IEEE Trans. Dependable Secure Comput. **9**(2), 222–235 (2012)
6. Davis, D., Monrose, F., Reiter, M.K.: On user choice in graphical password schemes. In: Proceedings of the 13th Conference on USENIX Security Symposium (SSYM), pp. 151–164. USENIX Association, Berkeley (2004)
7. De Luca, A., Hang, A., Brudy, F., Lindner, C., Hussmann, H.: Touch me once and i know it's you!: implicit authentication based on touch screen patterns. In: Proceedings of CHI, pp. 987–996. ACM (2012)
8. Dirik, A.E., Memon, N., Birget, J.C.: Modeling user choice in the passpoints graphical password scheme. In: Proceedings of the 3rd Symposium on Usable privacy and security (SOUPS), pp. 20–28. ACM, New York (2007)
9. Dunphy, P., Yan, J.: Do background images improve "draw a secret" graphical passwords? In: Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS), pp. 36–47 (2007)
10. Feng, T., et al.: Continuous mobile authentication using touchscreen gestures. In: Proceedings of the 2012 IEEE Conference on Technologies for Homeland Security (HST), pp. 451–456. IEEE (2012)
11. Fox, S.: Future Online Password Could be a Map (2010). http://www.livescience.com/8622-future-online-password-map.html
12. Frank, M., Biedert, R., Ma, E., Martinovic, I., Song, D.: Touchalytics: on the applicability of touchscreen input as a behavioral biometric for continuous authentication. IEEE Trans. Inf. Forensics Secur. **8**(1), 136–148 (2013)
13. Gołofit, K.: Click passwords under investigation. In: Biskup, J., López, J. (eds.) ESORICS 2007. LNCS, vol. 4734, pp. 343–358. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74835-9_23
14. Jermyn, I., Mayer, A., Monrose, F., Reiter, M.K., Rubin, A.D.: The design and analysis of graphical passwords. In: Proceedings of the 8th Conference on USENIX Security Symposium, pp. 1–14. USENIX Association, Berkeley (1999)
15. LIBSVM - A Library for Support Vector Machines. https://www.csie.ntu.edu.tw/~cjlin/libsvm/
16. Lin, D., Dunphy, P., Olivier, P., Yan, J.: Graphical passwords & qualitative spatial relations. In: Proceedings of the 3rd Symposium on Usable Privacy and Security (SOUPS), pp. 161–162 (2007)
17. Meng, Y.: Designing click-draw based graphical password scheme for better authentication. In: Proceedings of the 7th IEEE International Conference on Networking, Architecture, and Storage (NAS), pp. 39–48 (2012)

18. Meng, Y., Li, W.: Evaluating the effect of tolerance on click-draw based graphical password scheme. In: Chim, T.W., Yuen, T.H. (eds.) ICICS 2012. LNCS, vol. 7618, pp. 349–356. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-34129-8_32
19. Meng, Y., Li, W.: Evaluating the effect of user guidelines on creating click-draw based graphical passwords. In: Proceedings of the 2012 ACM Research in Applied Computation Symposium (RACS), pp. 322–327 (2012)
20. Meng, Y., Li, W., Kwok, L.-F.: Enhancing click-draw based graphical passwords using multi-touch on mobile phones. In: Janczewski, L.J., Wolfe, H.B., Shenoi, S. (eds.) SEC 2013. IAICT, vol. 405, pp. 55–68. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39218-4_5
21. Meng, W., Wong, D.S., Furnell, S., Zhou, J.: Surveying the development of biometric user authentication on mobile phones. IEEE Commun. Surv. Tutor. **17**(3), 1268–1293 (2015)
22. Meng, W.: RouteMap: a route and map based graphical password scheme for better multiple password memory. In: Qiu, M., Xu, S., Yung, M., Zhang, H., et al. (eds.) Network and System Security. LNCS, vol. 9408, pp. 147–161. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-25645-0_10
23. Meng, W.: Evaluating the effect of multi-touch behaviours on android unlock patterns. Inf. Comput. Secur. **24**(3), 277–287 (2016)
24. Meng, W., Li, W., Wong, D.S., Zhou, J.: TMGuard: a touch movement-based security mechanism for screen unlock patterns on smartphones. In: Manulis, M., Sadeghi, A.-R., Schneider, S. (eds.) ACNS 2016. LNCS, vol. 9696, pp. 629–647. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39555-5_34
25. Meng, W., Lee, W.H., Liu, Z., Su, C., Li, Y.: Evaluating the impact of juice filming charging attack in practical environments. In: Kim, H., Kim, D.-C. (eds.) ICISC 2017. LNCS, vol. 10779, pp. 327–338. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-78556-1_18
26. Meng, W., Fei, F., Li, W., Au, M.H.: Harvesting smartphone privacy through enhanced juice filming charging attacks. In: Nguyen, P., Zhou, J. (eds.) ISC 2017. LNCS, vol. 10599, pp. 291–308. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69659-1_16
27. Meng, W., Li, W., Kwok, L.-F., Choo, K.-K.R.: Towards enhancing click-draw based graphical passwords using multi-touch behaviours on smartphones. Comput. Secur. **65**, 213–229 (2017)
28. Meng, W., Li, W., Lee, W.H., Jiang, L., Zhou, J.: A pilot study of multiple password interference between text and map-based passwords. In: Gollmann, D., Miyaji, A., Kikuchi, H. (eds.) ACNS 2017. LNCS, vol. 10355, pp. 145–162. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61204-1_8
29. Meng, W., Lee, W.H., Au, M.H., Liu, Z.: Exploring effect of location number on map-based graphical password authentication. In: Pieprzyk, J., Suriadi, S. (eds.) ACISP 2017. LNCS, vol. 10343, pp. 301–313. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59870-3_17
30. Nelson, D.L., Reed, V.S., Walling, J.R.: Pictorial superiority effect. J. Exp. Psychol.: Hum. Learn. Mem. **2**(5), 523–528 (1976)
31. Nyang, D., et al.: Two-thumbs-up: physical protection for pin entry secure against recording attacks. Comput. Secur. **78**, 1–15 (2018)
32. Passfaces. http://www.realuser.com/
33. Quinlan, J.R.: Improved use of continuous attributes in C4.5. J. Artif. Intell. Res. **4**(1), 77–90 (1996)

34. Rennie, J.D.M., Shih, L., Teevan, J., Karger, D.R.: Tackling the poor assumptions of Naive Bayes text classifiers. In: Proceedings of the 20th International Conference on Machine Learning, pp. 616–623 (2003)
35. Rumelhart, D., Hinton, G., Williams, R.: Learning representations by back-propagating errors. Nature **323**, 533–536 (1986)
36. Shepard, R.N.: Recognition memory for words, sentences, and pictures. J. Verbal Learn. Verbal Behav. **6**(1), 156–163 (1967)
37. Smith-Creasey, M., Rajarajan, M.: A continuous user authentication scheme for mobile devices. In: Proceedings of the 14th Annual Conference on Privacy, Security and Trust (PST), pp. 104–113 (2016)
38. Spitzer, J., Singh, C., Schweitzer, D.: A security class project in graphical passwords. J. Comput. Sci. Coll. **26**(2), 7–13 (2010)
39. SplashData Inc., The Worst Passwords of 2018. https://www.teamsid.com/splashdatas-top-100-worst-passwords-of-2018/
40. Shahzad, M., Liu, A.X., Samuel, A.: Behavior based human authentication on touch screen devices using gestures and signatures. IEEE Trans. Mob. Comput. **16**(10), 2726–2741 (2017)
41. Sharma, V., Enbody, R.: User authentication and identification from user interface interactions on touch-enabled devices. In: Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec), pp. 1–11 (2017)
42. Suo, X., Zhu, Y., Owen, G.S.: Graphical passwords: a survey. In: Proceedings of the 21st Annual Computer Security Applications Conference (ACSAC), pp. 463–472. IEEE Computer Society (2005)
43. Sun, H., Chen, Y., Fang, C., Chang, S.: PassMap: a map based graphical-password authentication system. In: Proceedings of AsiaCCS, pp. 99–100 (2012)
44. Tao, H., Adams, C.: Pass-Go: a proposal to improve the usability of graphical passwords. Int. J. Netw. Secur. **2**(7), 273–292 (2008)
45. Thorpe, J., MacRae, B., Salehi-Abari, A.: Usability and security evaluation of GeoPass: a geographic location-password scheme. In: Proceedings of the 9th Symposium on Usable Privacy and Security (SOUPS), pp. 1–14 (2013)
46. Weka: Machine Learning Software in Java. https://www.cs.waikato.ac.nz/ml/weka/
47. Wiedenbeck, S., Waters, J., Birget, J.-C., Brodskiy, A., Memon, N.: Passpoints: design and longitudinal evaluation of a graphical password system. Int. J. Hum.-Comput. Stud. **63**(1–2), 102–127 (2005)
48. Weir, M., Aggarwal, S., Collins, M., Stern, H.: Testing metrics for password creation policies by attacking large sets of revealed passwords. In: Proceedings of CCS, pp. 162–175 (2010)
49. Yan, J., Blackwell, A., Anderson, R., Grant, A.: Password memorability and security: empirical results. IEEE Secur. Priv. **2**, 25–31 (2004)
50. Yang, Y., Guo, B., Wang, Z., Li, M., Yu, Z., Zhou, X.: BehaveSense: continuous authentication for security-sensitive mobile apps using behavioral biometrics. Ad Hoc Netw. **84**, 9–18 (2019)
51. Yu, X., Wang, Z., Li, Y., Li, L., Zhu, W.T., Song, L.: EvoPass: evolvable graphical password against shoulder-surfing attacks. Comput. Secur. **70**, 179–198 (2017)
52. Zhao, X., Feng, T., Shi, W., Kakadiaris, I.A.: Mobile user authentication using statistical touch dynamics images. IEEE Trans. Inf. Forensics Secur. **9**(11), 1780–1789 (2014)
53. Zheng, N., Bai, K., Huang, H., Wang, H.: You are how you touch: user verification on smartphones via tapping behaviors. In: Proceedings of the 2014 International Conference on Network Protocols (ICNP), pp. 221–232 (2014)