QUANTIFYING GRAPHICAL PASSWORD STRENGTH BY NEURAL NETWORKS

Yushuo Guan, Yuanxing Zhang, Lin Chen, Kaigui Bian*

Peking University, *Yale University

ABSTRACT

Graphical passwords have been widely used for protecting user privacy in many applications. Combining images and sketches in the design of graphical passwords could greatly increases the complexity for a regular user or a savvy shoulder-surfing attacker (that can record user inputs by video camera) to crack the password system. However, the recent advance in machine learning theory, e.g., the neural networks (NN), enables an attacker to make a better guess on the password by learning from user inputs, without the complete knowledge of the password system design. In this paper, we propose an NN-based framework of quantifying the graphical password strength against the shoulder-surfing attack. We present an NN-based attack model, and conduct experiments on four representative graphical password systems. Experimental results show that PIN and text passwords have serious security risks in presence of the shoulder-surfing attack, since the attack model can get a more than 90% precision with less than 10 training data.

***Index Terms*—** Graphical password, shoulder-surfing attack, neural network

1. INTRODUCTION

People are increasingly using graphical passwords to login their online accounts and access personal information (e.g., to visit social networks like Facebook or Twitter). Graphical passwords are knowledge-based passwords, motivated by the idea that correlates images or sketches with the shared secret [1]. The simplest implementations of graphical passwords are the Personal Identification Number (PIN) and text passwords, which directly use numbers and characters to create a password. More advanced passwords add more types of symbols in the password design, like human faces or photos [2, 3, 4].

The use of images and sketches in the password design improves the theoretical complexity for a regular user or a savvy shoulder-surfing attacker (that can record user inputs by video camera) to crack the password system. The shoulder-surfing attack, a.k.a. peeping attack, allows adversaries to observe all actions of humans on input terminals and interactions between humans (provers) and computers (verifiers) [5, 6]. Attackers may utilize direct observation or high-resolution cameras to record passwords. The theoretical complexity of guessing the password will decrease quickly after a number of successive observations by the shoulder surfing attacker. For example, the theoretical complexity of guessing a text password will decrease to $O(1)$, if the attacker successfully observes the user input for only one-time.

The vulnerability of text password is that the user input is exactly equal to the password itself. That is why the password could be exposed when the shoulder-surfing attacker captures the user input for one time. Hence, to defeat the shoulder surfing attack, many advanced graphical password systems seek to create a mapping function between the password and the user input—the user input is no longer equal to the password. The user has to do the mapping by translating the password in his/her mind to a user input on the keypad. As a result, it is largely insufficient for attackers to crack the password merely with the records of user input.

However, the advance in neural network (NN) allows the attacker with shoulder-surfing methods to make a better guess on the graphical password by learning the possible mapping functions between the user input on the keypad and the password in the user's mind. The advantage of the NN model is that it does not require hand-crafted features but it can directly learn from the raw observations. More importantly, NN makes it possible to quantify the strength of passwords against shoulder-surfing attacks. Conventional qualitative analysis on password strength sends certain questionnaires to and collect feedbacks from a group of participants, which is coarse-grained and bias-prone.

Recent work has focused on the invention of shoulder-surfing resistant authentication systems. Zhang et al. [7] and Maiti et al. [8] make shoulder-surfing resistant schemes for augmented reality; Luo and Yang [9] and Higashiyama et al. [10] present authentication schemes for smartphones to prevent shoulder-surfing attacks. All these systems are evaluated by a small number of volunteers, which is less convincing than a quantitative analysis. In contrast, NN evaluates the robustness of the graphical password by the theoretical complexity without requiring the knowledge of the password design. Hitaj et al. [11] and Melicher et al. [12] use generative adversarial networks [13] and recurrent neural networks to crack passwords, but they only focus on guessing attacks and their models merely apply to text password systems.

In this paper, we propose a NN-based framework for ana-

lyzing the password strength against shoulder-surfing attack. We redefine a graphical password system in the new framework, and propose an indicator to measure password strength against the shoulder surfing attack. We then establish different NN models for graphical passwords with single or multiple bits of input. We verify the rationality of the proposed indicator of password strength and study the impact of NN's information expressivity on the indicator. We have released the source code and relative datasets on GitHub[1].

2. SYSTEM MODEL

In this section, we first present basic definitions with respect to an authentication, and then introduce the architecture in the graphical password systems.

2.1. Basic Definitions

The two fundamental components in the authentication system are authentication challenge and answer [14]. To login, users have to enter an answer into the system (like "123456") based on a given challenge (like "Please enter your password.").

Definition 1 ***Authentication challenge*** $\vec{c}$ ***and challenge space*** $\mathbb{C}$. $\vec{c} = [x_1, x_2, ..., x_M]$ *contains* M*-bit messages for computers (verifiers) to distinguish users from malicious impersonators. Challenge space contains all possible authentication challenges that one graphical password system can provide.* $|\mathbb{C}| = \prod_{i=1}^{M} |S_x^i|$*, where* S_x^i *contains all possible options for the* i*-th bit message of an authentication challenge.*

Definition 2 ***Authentication answer*** $\vec{a}$ ***and answer space*** $\mathbb{A}$. *For an* N*-bit graphical password system,* $\vec{a} = [y_1, y_2, ..., y_N]$*, where* y_i *is the* i*-th input of users in the authentication process.* $(i \in [1, N], i \in \mathbb{N})$ *The answer space contains all possible authentication answers a certain user can provide:* $|\mathbb{A}| = \prod_{i=1}^{N} |S_y^i|$*, where* S_y^i *contains all possible options for the* i*-th input of an answer.*

Given the definitions of authentication challenge and authentication answer, we can propose the definition of the authentication password.

Definition 3 ***Authentication password*** p ***and password space*** $\mathbb{P}$. *The passwords can be viewed as a mapping from challenge space to answer space:* $p : \mathbb{C} \to \mathbb{A}$. *For a specific authentication challenge* $\vec{c} \in \mathbb{C}$*,* $p(\vec{c})$ *is the correct authentication answer. Password space contains all possible authentication passwords.* $|\mathbb{P}| = |\mathbb{A}|^{|\mathbb{C}|}$.

[1] https://github.com/PkuDavidGuan/graphical-password

2.2. Architecture of the graphical password system

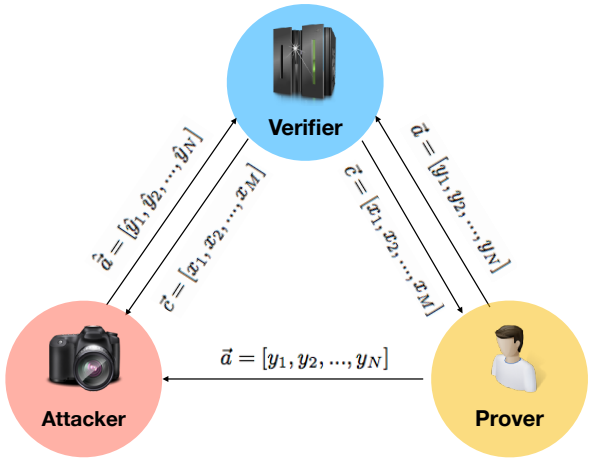

Fig. 1. The architecture of a graphical password systems.

As shown in Figure 1, a graphical password system is composed of a verifier (computer), K provers (human users), and an attacker.

The verifier is responsible for the verification of provers. It maintains $[p_1, p_2, ..., p_K]$, which p_i is the authentication password of prover i. In the authentication process, it generates K authentication challenges $[\vec{c}_1, \vec{c}_2, ..., \vec{c}_K]$ for each prover, and collects replies $[\vec{a}_1, \vec{a}_2, ..., \vec{a}_K]$ from each prover. The verifier distinguishes true provers $\mathbb{T}$ from attackers, where $\mathbb{T} = \{i | p_i(\vec{c}_i) = \vec{a}_i\}$.

For prover i, it gets an authentication challenge $\vec{c}_i$ from the verifier and replies with an authentication answer $\vec{a}_i = p_i(\vec{c}_i)$. Prover i can login only if $i \in \mathbb{T}$.

The attacker hides himself in the system, records successive challenge-answer pairs $(\vec{c}, \vec{a})$, and trains the NN-based attack models against different provers. The result of an attack is determined by Equation (1), and $\mathcal{A}(\vec{c}_i, \vec{a}_i) = 1$ will be viewed as a successful attack on prover i.

$$\mathcal{A}(\vec{c}, \vec{a}) = \begin{cases} 1, & NN(\vec{c}) = \vec{a} \\ 0, & NN(\vec{c}) \neq \vec{a} \end{cases} \tag{1}$$

3. DESIGN OF THE NN-BASED ATTACK MODEL

In this section, we describe the design of an NN-based attack model.

3.1. The training procedure

The first step for the attack model is to generate a training phase. Ideally, training data could be collected by the shoulder-surfing attackers. However, this approach is timconsuming, as the training algorithm needs to wait until an

attack is complete. To accelerate the process, we train the attack model by simulations that faithfully emulate the authentication process. The simulator maintains a verifier and a prover, which could generate authentication challenges and relevant answers successively.

3.2. The NN-based attack model

The input of the model is an M-dimensional vector $\vec{c} = [x_1, x_2, ..., x_M]$, which an authentication challenge. The output is simply a predicted N-bit answer with respect to the authentication challenge.

The input layer is simply M units, i.e. one for each bit of an authentication challenge. The hierarchy of hidden layers is optional for attackers. The output layer differs between password designs.

- For the prediction of the i-bit authentication answer, if S_y^i is known by attackers in advance, we will use a $|S_y^i|$-way softmax as predict the i-bit of the authentication answer.
- Otherwise, the softmax layer will be replaced by a regression layer.

The i-bit authentication answer can be predicted as:

$$\begin{aligned} a_i = (1 - \mathcal{K}(S_y^i))(\sum_{i=1}^{H} w_i x_i) \\ +\mathcal{K}(S_y^i)(\arg\max_{k \in S_y^i} \frac{exp(x_k)}{\sum_{j=1}^{S_y^i} exp(x_j)}) \end{aligned} \tag{2}$$

where H is the size of the last hidden layer, x_i is the output of the i-th unit in the last hidden layer. $\mathcal{K}(S_y^i) = 1$ when S_y^i is known in advance, otherwise $\mathcal{K}(S_y^i) = 0$.

The NN-based attack model only focuses on the high precision, which means attackers could pass the authentication with a high probability. The characteristics of the NN-based attack model distinguish itself from the conventional NN model, as the latter pay equal attention to precision and recall rates.

3.3. Password strength indicator

With the NN-based attack model, we could quantify passwords strength against shoulder-surfing attacks from attackers' view, since the difficulty of NN training indirectly reflects password strength. Given the training set R_{train} and test set R_{test}, we define an indicator to quantify the strength of password system $pass$ against the shoulder-surfing attack:

$$\begin{aligned} \mathcal{S}_{pass|l} = \min \ \ |R_{train}| \\ s.t. \ \forall (c, a) \in R_{test}, P(\mathcal{A}(c, a) = 1 | R_{train}) \geq l \end{aligned} \tag{3}$$

The indicator $\mathcal{S}_{pass|l}$ is highly influenced by the information expressivity of NN models. Therefore, for the rationality of the comparisons of password strength, the information expressivities of all NN models should be similar when $\mathcal{S}_{pass|l}$ is used to quantify multiple password systems at the same time.

4. EVALUATION

In this section, we first introduce two categories of advanced graphical password systems, and present our dataset and model architectures. We then conduct two experiments and study the rationality of the indicator $\mathcal{S}_{pass|l}$, and the impact of NN models' expressivity. The code of the experiment is available on Github (see [15] for details).

4.1. Experiment setup

Advanced graphical password systems: Advanced graphical passwords can be divided into two main categories based on recall and recognition [16].

Recognition-based systems generally ask users to memorize a portfolio of images during password creation and then recognize their images from among decoys [1, 17]. PassFaces [2] is the canonical example of recognition-based graphical passwords. PassFaces has a pool of F human face pictures, where users pre-select A human faces. During login, a panel of B candidate faces is presented. Users must select the face belonging to their set from among decoys. A number of C such rounds are repeated with different panels (generally $C < A$).

Recall-based graphical password systems are referred to as drawmetric systems [18]. In the recall-based systems, users recall the pattern they selected before and calculate an authentication answer based on the pattern. GrIDsure [19] is a kind of recall-based graphical password system, which displays digits in a $D \times D$ grid. Users select and memorize a pattern consisting of an ordered subset of E grid squares in advance. On subsequent logins, digits are randomly displayed within the grid squares, and users enter the sum of digits within their memorized pattern.

Setup: We conduct experiments on 6-bit PIN, indefinite length (≤ 12) text password, PassFaces, and GrIDsure. Note that the 6-bit PIN has been widely used in ATM and online payment environments, and indefinite length text passwords are applied in social applications, like Facebook and Twitter. These are two representative types of conventional graphical password systems. In the experiments, we set the following parameters $A = 5, B = 9, C = 1, D = 3, E = 4, F = 100$.

Attack model: We use Keras [20] to construct attack models for these four graphical passwords. There are two hidden layers for all attack models, and each layer has an identical number of hidden units. Identical hidden layers ensure that the four NN models have similar information expressivity. The number of units in the input and output layers is determined by the size of authentication challenges, and answers

of different password systems, which makes little influence on the information expressivity. In the training process, we train each NN model with 10 different training sets, and compute the average precision. We use RMSProp as the optimizer.

4.2. Experimental results

Rationality: In the first experiment, we seek to verify the rationality of our indicator $\mathcal{S}_{pass|l}$. All the NN models in this experiment have two hidden layers, and each layer has 32 hidden units. We record the precision rate of four attack models, given different sizes of training sets.

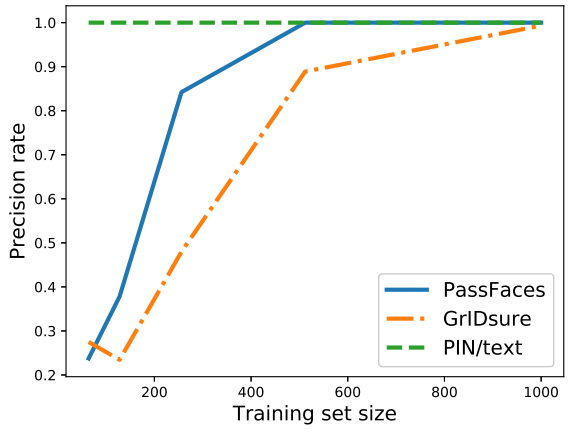

Fig. 2. The precision of different graphical passwords, given different sizes of the training set.

The result is shown in Figure 2. To achieve a 90% precision or higher, the NN models for PIN/text need really small training set, the NN model for PassFaces need 256 pieces of training data, and about 500 pieces of training data are needed for GrIDsure. The result shows that GrIDsure and PassFaces have a better robustness than PIN/text passwords against the shoulder-surfing attack, and GrIDsure is even better than PassFaces.

The experimental result is consistent with the analytical result. The challenge space $\mathbb{C}$, and the answer space $\mathbb{A}$ of PIN and text passwords are zero-dimension. So, the authentication passwords in these systems are constant functions, which are vulnerable to the shoulder-surfing attack. For PassFaces, $|\mathbb{C}| = \binom{F}{B}$, $|\mathbb{A}| = \binom{A}{C}$, and the authentication password is an injective function, which is more difficult for NN-based models to crack. GrIDsure has an even larger challenge space and answer space, so it is less vulnerable to the attack.

The result also reveals the high vulnerability of PIN and text passwords in the presence of the shoulder-surfing attack. Although these two traditional password systems are widely adopted in mobile devices, which may lead to severe security risks.

Comparisons of multiple NN models: In the second experiment, we build up several NN models with different information expressivities, and study their influence on the indicator $\mathcal{S}_{pass|l}$.

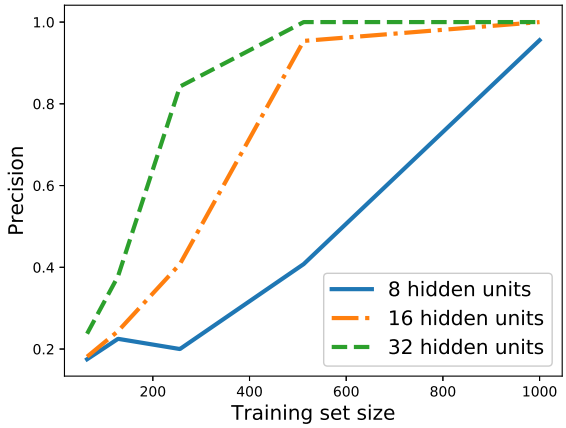

Fig. 3. The impact of information expressivity on PassFaces's NN model.

Figure 3 shows the results on PassFaces and GrIDsure. When there are 8 hidden units in each hidden layer, $\mathcal{S}_{PassFaces|0.9}$ approaches 1000. However, $\mathcal{S}_{PassFaces|0.9}$ halves when the number of hidden units increases to 32. The result on GrIDsure is similar, which is omitted here. The attack model for GrIDsure cannot get a 90% accuracy, even with 1000 pieces of training data when there are 8 hidden units each layer.

The second experiment shows that information expressivities of NN models can seriously affect the password strength $\mathcal{S}_{pass|l}$. The result is significant both for the attacker and researchers: the shoulder-surfing attacker will be more efficient when using state-of-the-art NN models; meanwhile, for researchers, the architecture of the NN attack model must be specified before quantifying a password system with $\mathcal{S}_{pass|l}$.

5. CONCLUSION

In this paper, we propose a neural network based framework for quantifying the graphical password strength against the shoulder-surfing attack. We cast the basic definitions of password systems into the framework, and present an NN-based attack model. We then conduct experiments on four representative graphical password systems; and study the rationality of the proposed indicator, as well as the impact of NN's information expressivity. We also notice that a more than 90% precision can be achieved for the attack model to crack PIN/text passwords, with less than 10 pieces of training data. The result indicates that PIN and text passwords have serious security risks in presence of the shoulder-surfing attack.

6. REFERENCES

[1] Robert Biddle, Sonia Chiasson, and Paul C. van Oorschot, "Graphical passwords: Learning from the first twelve years," *ACM Comput. Surv.*, vol. 44, no. 4, pp. 19:1–19:41, 2012.

[2] Passfaces Corporation, "The science behind passfaces," http://www.passfaces.com/enterprise/resources/, 2009.

[3] Ian Jermyn, Alain J. Mayer, Fabian Monrose, Michael K. Reiter, and Aviel D. Rubin, "The design and analysis of graphical passwords," in *Proceedings of the 8th USENIX Security Symposium, Washington, D.C., August 23-26, 1999*, 1999.

[4] Susan Wiedenbeck, Jim Waters, Jean-Camille Birget, Alex Brodskiy, and Nasir D. Memon, "Authentication using graphical passwords: effects of tolerance and image choice," in *Proceedings of the 1st Symposium on Usable Privacy and Security, SOUPS 2005, Pittsburgh, Pennsylvania, USA, July 6-8, 2005*, 2005, pp. 1–12.

[5] Shujun Li and Heung-Yeung Shum, "Secure human-computer identification (interface) systems against peeping attacks: Sechci," *IACR Cryptology ePrint Archive*, vol. 2005, pp. 268, 2005.

[6] Ross J. Anderson, "Why cryptosystems fail," *Commun. ACM*, vol. 37, no. 11, pp. 32–40, 1994.

[7] Ruide Zhang, Ning Zhang, Changlai Du, Wenjing Lou, Y. Thomas Hou, and Yuichi Kawamoto, "Augauth: Shoulder-surfing resistant authentication for augmented reality," in *IEEE International Conference on Communications, ICC 2017, Paris, France, May 21-25, 2017*, 2017, pp. 1–6.

[8] Anindya Maiti, Murtuza Jadliwala, and Chase Weber, "Preventing shoulder surfing using randomized augmented reality keyboards," in *2017 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2017, Kona, Big Island, HI, USA, March 13-17, 2017*, 2017, pp. 630–635.

[9] Jia-Ning Luo and Ming-Hour Yang, "A mobile authentication system resists to shoulder-surfing attacks," *Multimedia Tools Appl.*, vol. 75, no. 22, pp. 14075–14087, 2016.

[10] Yuma Higashiyama, Naoto Yanai, Shingo Okamura, and Toru Fujiwara, "Revisiting authentication with shoulder-surfing resistance for smartphones," in *Third International Symposium on Computing and Networking, CANDAR 2015, Sapporo, Hokkaido, Japan, December 8-11, 2015*, 2015, pp. 89–95.

[11] Briland Hitaj, Paolo Gasti, Giuseppe Ateniese, and Fernando Pérez-Cruz, "Passgan: A deep learning approach for password guessing," *CoRR*, vol. abs/1709.00440, 2017.

[12] William Melicher, Blase Ur, Sean M. Segreti, Saranga Komanduri, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor, "Fast, lean, and accurate: Modeling password guessability using neural networks," in *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016.*, 2016, pp. 175–191.

[13] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio, "Generative adversarial networks," *CoRR*, vol. abs/1406.2661, 2014.

[14] Jeremiah Blocki, Manuel Blum, and Anupam Datta, "Human computable passwords," *CoRR*, vol. abs/1404.0024, 2014.

[15] Yushuo Guan et al., "Graphical passwords," https://github.com/PkuDavidGuan/graphical-password, 2017.

[16] J. G. Raaijmakers and R. M. Shiffrin, *Models for recall and recognition.*, University Park Press,, 1992.

[17] Karen Renaud, "Guidelines for designing graphical authentication mechanism interfaces," *IJICS*, vol. 3, no. 1, pp. 60–85, 2009.

[18] Antonella De Angeli, Lynne M. Coventry, Graham Johnson, and Karen Renaud, "Is a picture really worth a thousand words? exploring the feasibility of graphical authentication systems," *International Journal of Man-Machine Studies*, vol. 63, no. 1-2, pp. 128–152, 2005.

[19] Sacha Brostoff, Philip Inglesant, and M. Angela Sasse, "Evaluating the usability and security of a graphical one-time pin system," in *Bcs Interaction Specialist Group Conference*, 2010, pp. 88–97.

[20] François Chollet et al., "Keras," https://github.com/fchollet/keras, 2015.