TwoStep: An Authentication Method Combining Text and Graphical Passwords

P.C. van Oorschot   Tao Wan [⋆]

School of Computer Science, Carleton University, Ottawa, Canada
{*paulv, twan*}*@scs.carleton.ca*

Abstract. *Text-based passwords alone are subject to dictionary attacks as users tend to choose weak passwords in favor of memorability, as well as phishing attacks. Many recognition-based graphical password schemes alone, in order to offer sufficient security, require a number of rounds of verification, introducing usability issues. We suggest a hybrid user authentication approach combining text passwords, recognition-based graphical passwords, and a two-step process, to provide increased security with fewer rounds than such graphical passwords alone. A variation of this two-step authentication method, which we have implemented and deployed, is in use in the real world.*

Keywords: Graphical Passwords, User Authentication, Phishing, Security

1 Introduction

Text passwords have been widely used for user authentication, e.g., by almost all websites on the Internet. However, it is well-known that text passwords are insecure for a variety of reasons. For example, users tend to choose simple passwords in favour of memorability, making them subject to dictionary attacks; and text passwords can be stolen by malicious software (e.g., keystroke loggers) when being entered from keyboards. Phishing is another serious threat to text passwords, by which, a user could be persuaded to visit a forged website and enter their passwords. Such an attack is made possible in part due to the fact that text passwords do not allow users to authenticate a server; by design they provide only one-way user authentication, and server authentication is not a design objective of text passwords alone.

We propose a two-step authentication method to strengthen text passwords by combining them with graphical passwords. In this approach, called *TwoStep*, users continue to use text passwords as a first step, but then must also enter a graphical password, providing the following advantages: (1) users' current sign-in experience is largely preserved; (2) a text password alone which is stolen (e.g., by phishing) does not compromise an account; (3) users can be alerted if not seeing the graphical password cuing image after providing their text passwords, implicitly providing server authentication; and (4) it can be implemented in software alone, increasing the potential for large-scale adoption on the Internet.

[⋆] Corresponding author.

The rest of this paper is organized as follows. In Section 2, we describe TwoStep, and consider its security. Section 3 provides preliminary security analysis for TwoStep. Section 4 briefly reviews related work. We conclude in Section 5.

2 Two-step Authentication Method

Given that text passwords are easy to deploy and to use, we believe that they will continue to be popular. Thus, we suggest that effort should be made to enhance text passwords with an easy to use additional defense mechanism that can address common password attacks, such as brute-force and phishing attacks. To this end, we propose *TwoStep*, a combination of text passwords and recognition-based graphical passwords. The latter can complement text passwords being less subjective to phishing attacks which require prior knowledge of users' image portfolios, and to naive keylogger attacks.

In step one, a user is asked for her user name and text password. After supplying this, and independent of whether or not it is correct, in step two, the user is presented with an image portfolio. The user must correctly select all images (one or more) pre-registered for this account in each round of graphical password verification. Otherwise, account access is denied despite a valid text password. Using text passwords in step one preserves the existing user sign-in experience. If the user's text password or graphical password is correct, the image portfolios presented are those as defined during password creation. Otherwise, the image portfolios (including their layout dimensions) presented in first and a next round are random but respectively a deterministic function of the user name and text password string entered, and the images selected in the previous round. More specifically, the image portfolio in round n is pseudo-randomly generated from a seed value derived from the entered user name and text password when $n = 1$, and from the images selected in round $n - 1$ when $n \geq 2$.

Seeing a portfolio including no familiar image allows a legitimate user to immediately realize that she entered an invalid text or graphical password (and then go back to re-enter it, e.g., using a "Go Back" dialog button), but prevents an attacker from knowing that the text or graphical password tried is invalid (cf. [3, 2]).

Creation of Graphical Passwords. Graphical passwords can be created during user registration or after registration (for users registerred before TwoStep was implemented), and be changed any time after creation. A graphical password policy, which may be set by the site operator or the user, influences its presentation and security. Example policy attributes are: *number of rounds of verification*; *display layout*, e.g., 6×6, defining how images are presented to the user, and the total number of images displayed in each round; *number of images* to be selected in each round; and *ordered* or *unordered* image selection, defining whether order of image selection matters.

After a graphical password policy is defined, users choose images as their graphical passwords. For each round of verification, the specified number of images are randomly selected by the system from a database to form an image portfolio. A user then chooses a specified number of images from the portfolio as her graphical password components. This process repeats for the specified number of rounds. If the user does not like a particular image portfolio, she may request a new one or upload her own images to be included in a portfolio. An accepted image portfolio remains unchanged until the user changes her graphical password. To facilitate recognition, images within a portfolio are assembled to be sufficiently distinguishable.

Subsequent Login Using TwoStep. In step one, the user as usual enters a user name and text password. The login page of the server deploying TwoStep remains the same as when text passwords alone were used, i.e., no change in the front login page is required to deploy TwoStep, nor do users see any difference in their sign-in experience in step one. After the user provides a text password, the second step of authentication (the graphical step or g-step) begins. In each round of graphical password verification, the server transmits an image portfolio to the user, and the user chooses out her pre-registered images. After the user completes all rounds of verification, if both the text password and all graphical passwords were correct, she is granted account access. Otherwise, access is denied. We next discuss several attacks against graphical passwords which must be considered. Further security discussion is found in Section 3.

Eavesdropping. An attacker able to intercept communication between the server and client would be able to capture image portfolios transmitted from the server, and the images selected by the user, thus stealing the entire graphical password. To prevent this attack, a security protocol such as HTTPS must be deployed to provide confidentiality.

Shoulder-Surfing. An attacker can also steal a graphical password by shoulder-surfing (e.g., using a video camera) during the g-step. Such shoulder-surfing would be particularly easy if an implementation of the g-step provided user visual feedback upon user selection of an image, such as highlighting an image border. Here we describe a simple method to mitigate this type of attack (see Fig. 1).

For a given image portfolio, each image is associated with an index number. Images along with their index numbers are displayed in a random order on the screen. Below the displayed image portfolio is a *selection panel* with all index numbers displayed incrementally. To select an image, the user identifies the image and then clicks the corresponding index number on the lower selection panel. In the case that several images must be chosen from a portfolio, the selection panel can help the user keep track of which images have been selected so far (and allows easy de-selection, by clicking the corresponding number in the bottom panel, if necessary). The idea is that it is more difficult for a casual human observer to have line of sight to the lower panel and to map an index or set of indices from it to the corresponding images on the screen. This approach can reduce casual shoulder-surfing but cannot fully prevent such attacks involving movie-clip camera phones. Other techniques, e.g., Gaze-based password entry [13], can better mitigate this type of attack, but have their own usability and deployment challenges.

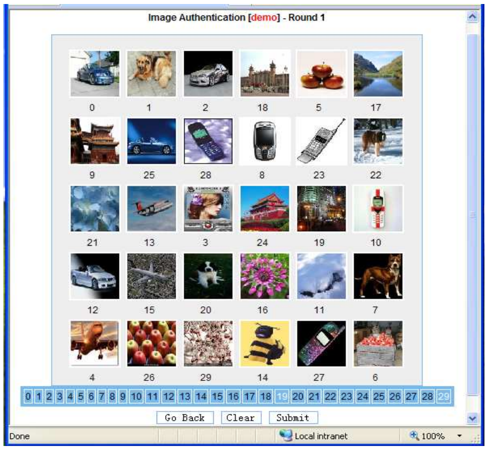

Fig. 1. Selection panel in graphical step

3 Preliminary Security Analysis

Password Strength. We discuss the strength of TwoStep, measured by entropy in bits, by considering both the entropy of the text password and the graphical password parts. A text password of length l characters has entropy of $l \cdot \log_2 c$ bits if characters are selected uniformly at random and independently from an alphabet of c characters. For example, a randomly generated 8-character password consisting of digits, lowercase, and uppercase has $8 \cdot \log_2 62 = 47.6$ bits of entropy.

Let r be the number of rounds of our graphical password verification. For each round, let n be the size of the image portfolio, and $k < n$ the total number of images selected from the portfolio as the graphical password. The entropy of a randomly selected graphical password conforming to this policy is $r \cdot \log_2 t$, where $t = \binom{n}{k}$, and $\frac{n!}{(n-k)!}$ for un-ordered and ordered images respectively.

As an example, consider $r = 1$, $n = 36$, $k = 3$, and unordered image selection, meaning one round of verification by selecting 3 images in any order from a portfolio of size 36. The entropy is $\log_2 \binom{36}{3} \approx 12.8$ bits. For $r = 2$, in theory this doubles to 25.6 bits, though in practice we might expect less unpredicatability due to patterns in user choice [5]. Choosing different parameters k, n, r, t can increase security, but also changes usability. In addition, password guessing attacks in TwoStep must be done online (interacting with the server), which is more costly than offline attacks.

Note that text passwords used in practice are generally far from randomly and independently selected, and often lowercase only (cf. [10]), decreasing entropy. For example, an 8-character lowercase password has entropy about 37.6 bits if all characters were selected randomly and independently. But in practice, they perhaps have only 20-35 bits on average and less for some subsets of users. Relative to this more realistic estimate, the 25.6 bits (or even 12.8 bits) of added security from the graphical part is quite significant, against both targeted single-account exhaustive attacks, and system-wide multi-account attacks that might attempt as few as 3-5 guesses per account.

Mitigating Naive Keylogging Attacks. Keylogging is a common method for stealing user text passwords. A keylogger is malicious software which intercepts keystrokes on an infected machine as a user types. For example, Microsoft Windows provides (undocumented) interfaces facilitating interception of system events including keystrokes. With TwoStep, a user would use the keyboard for the text password part, and mouse clicks for the graphical parts. Thus, a naive keylogger cannot obtain the graphical parts. More sophisticated malware can capture both user screen contents and mouse clicks to recover a graphical password, with more effort.

Mitigating Phishing Attacks. Phishing [7] is another common technique for stealing passwords by fooling users to enter such information into a fraudulent website spoofing a legitimate one (e.g., a bank site). Social engineering tactics are often used (e.g., “urgent account update”, requests to verify fake transactions, etc.). In TwoStep, while users’ text password part can still be stolen by phishing, obtaining their graphical password parts is more difficult: without knowledge of users’ image profiles, the phisher does not know what images to present in order to extract a graphical password.

Mitigating Active MITM Attacks. An active man-in-the-middle (MITM) attack allows an attacker to become an intervening proxy and control all communication between the user and the website (cf. [9]). SSL cannot mitigate this attack since an attacker can use SSL on both communication segments individually, so users (and end website) appear to be "operating securely". The proxy can be either malware on a user's local machine or located on a remote server (controlled by an attacker) to which the user is drawn by phishing techniques. Such an attacker can gain access to any information exchanged between a user and a website, thus can defeat TwoStep.

It appears difficult to prevent this active MITM attack if the end-user machine is infected by malicious software. In fact, it seems all software-only defenses fail for such compromised end-machines. On the other hand, if the active MITM proxy is located remotely, as in DNS server pharming-based MITM attacks, consistency check techniques involving alternative communication paths could be used to detect if requests intended to be sent to one server actually terminate at another. This provides protection to TwoStep against active MITM proxies.

Comparison with Challenge Questions. *Challenge questions* [14] are now widely used for recovering or resetting forgotten passwords, as well as authenticating users. For example, Royal Bank of Canada allows users to register a primary computer with the bank for online banking by accepting a cookie on that machine. When a user signs in from this computer, she will be authenticated by her account number and password. When she signs in from a non-primary computer, authentication involves these, as well as a challenge question. ING Direct in Canada also uses challenge questions, but in contrast, prior to passwords. A user first provides her account number, then answers a challenge question. After a correct answer, a personalized site image is displayed, and the user is asked to enter her password. The user is supposed to enter her password only when seeing the site image, supposedly protecting the user password. One advantage of TwoStep over a combination of challenge questions and text passwords (both entered from keyboard) is that it is less vulnerable to naive keylogging attacks, because the graphical password part in the former is entered by mouse clicks.

4 Related Work

Graphical passwords can be largely classified into three categories: recognition-based, cued-recall, or recall-based. In recognition-based graphical passwords, users are required to recognize and then select a set of preselected images from a larger set. In cued-recall, the images cue the user, for example, to click a set of points on an image [3]. In recall-based, users are required to recall a password without any cues, such as drawing a doodle in Draw-A-Secret (DAS) [12]. We focus the remainder of our review here on recognition-based schemes. For a broader survey, see Chiasson [1, Chapter 2].

Deja Vu [6] is a recognition-based graphical password, which makes use of random art images, instead of photographs, to discourage users from selecting predictable images. While randomly generated images can improve security, they also reduce usability. For example, it takes longer for users to remember random art images than photos, and less time to forget them. Passfaces [4] is another recognition-based scheme, using human faces as authentication images. A user's password consists of k faces, each of which must be chosen from a set of $n > 1$ faces in each round of the selection. While human faces are more memorable than text passwords, it was also found [5] that users usually choose predictable faces as their passwords, e.g., faces of their own race. In addition, female faces and "attractive" faces are chosen more often than male faces. Those biases make human faces less suitable as password components.

Story [5] is similar to Passfaces, but uses a variety of photos to form image portfolios, and encourages users to select photos to form a story to improve memorability. In Weinshall's scheme [15], a user is asked to answer a sequence of questions based on a shared set of images with the server. This scheme can resist shoulder-surfing attacks, but requires significant training and has usability issues, as well as security issues [11].

5 Concluding Remarks

TwoStep offers some advantages in countering common attacks against text passwords, such as naive keylogging and phishing. We have implemented [8] a variation of TwoStep (including the selection panel) as an optional mechanism for protecting online backup of user data in a Windows-based password manager, and it has been chosen and used on a regular basis by more than 4000 users, suggesting that a combination of text and graphical passwords is usable. An obvious and necessary next step is a user study, ideally both a lab study and a field study leveraging our real-world deployment.

Acknowledgements

Funding through an NSERC Discovery Grant, Canada Research Chair, and NSERC ISSNet is gratefully acknowledged.

References

1. S. Chiasson. *Usable Authentication and Click-Based Graphical Passwords*. PhD thesis, Carleton University, Ottawa, Canada, January 2009.
2. S. Chiasson, A. Forget, R. Biddle, and P.C. van Oorschot. Influencing Users Towards Better Passwords: Persuasive Cued Click-Points. In *Proc. of HCI'08*, September 2008.
3. S. Chiasson, P.C. van Oorschot, and R. Biddle. Graphical Password Authentication Using Cued Click Points. In *Proc. of ESORICS'07*, volume 4734, pages 359–374, September 2007.
4. Real User Corporation. The Science Behind Passfaces, September 2001.
5. D. Davis, F. Monrose, and M. Reiter. On User Choice in Graphical Password Schemes. In *Proc. of* 13^{th} *USENIX Security Symposium*, August 2004.
6. R. Dhamija and A. Perrig. Deja Vu: A User Study Using Images for Authentication. In *Proc. of* 9^{th} *USENIX Security Symposium*, August 2000.
7. R. Dhamija, J. Tygar, and M. Hearst. Why Phishing Works. In *Proc. of Human Factors in Computing Systems*, April 2006.
8. 51Logon: Simplifying SignIn Experience. `http://www.51Logon.com` (in Chinese).
9. E. Felton, D. Balfanz, D. Dean, and D. Wallach. Web Spoofing: An Internet Con Game. In *Proc. of the* 20^{th} *National Information systems Security Conference*, October 1997.
10. D. Florencio and C. Herley. A Large-Scale Study of Web Password Habits. In *Proc. of the 2007 World Wide Web*, 2007.
11. P. Golle and D. Wagner. Cryptanalysis of a Cognitive Authentication Schemes (Extended Abstract). In *Proc. of the 2007 IEEE Symposium on Security and Privacy*, May 2007.
12. I. Jermyn, A. Mayer, F. Monrose, M.K. Reiter, and A. Rubin. The Design and Analysis of Graphical Passwords. In *Proc. of the* 8^{th} *USENIX Security Symposium*, August 23-26 1999.
13. M. Kumar, Tal Garfinkel, D. Boneh, and T. Winograd. Reducing Shoulder-surfing by Using Gaze-based Password Entry. In *Proc. of SOUPS'07*, July 2007.
14. A. Rabkin. Personal Knowledge Questions for Fallback Authentication. In *Proc. of the 2008 Symposium On Usable Privacy and Security (SOUPS)*, July 23-25 2008.
15. D. Weinshall. Cognitive Authentication Schemes Safe Against Spyware (Short Paper). In *Proc. of the 2006 IEEE Symposium on Security and Privacy*, May 2006.