**International Journal of Emerging Trend in Engineering and Basic Sciences (IJEEBS)**
*ISSN (Online)* 2349-6967
*Volume 2 , Issue 1(Jan-Feb 2015), PP368-374*

# Captcha As A Graphical Password Using Click Event

[1]Sanket Bhor, [2]Hitesh Patil, [3]Sujata Ahire
*Students,S.G.O.I.C.O.E , Belhe, Pune .*

**Abstract**:- Current authentication systems suffer from many weaknesses. Textual passwords are commonly used; however, users do not follow their requirements. Users tend to choose meaningful words from dictionaries, which make textual passwords easy to break and vulnerable to dictionary or brute force attacks. Many available graphical passwords have a password space that is less than or equal to the textual password space. Smart cards or tokens can be stolen. Many biometric authentications have been proposed; however, users tend to resist using biometrics because of their intrusiveness and the effect on their privacy. Moreover, biometrics cannot be revoked. In this project, we present and evaluate our contribution, a novel family of graphical password systems built on top of Captcha technology, which we call Captcha as graphical passwords (CaRP). CaRP is click-based graphical passwords, where a sequence of clicks on an image is used to derive a password. Unlike other click-based graphical passwords, images used in CaRP are Captcha challenges, and a new CaRP image is generated for every file downloading. CaRP also offers protection against relay attacks, an increasing threat to bypass Captcha as protection, wherein Captcha challenges are relayed to humans to solve.. CaRP is robust to shoulder-surfing attacks if combined with dual-view technologies. CaRP requires solving a Captcha challenge in every file downloading.
**Keywords: -** Captcha, CaRP, Graphical Password, Password, security Analysis, Recognition recall CaRp, Recognition based CaRP.

## I. INTRODUCTION

CAPTCHA (Completely Automatic Public Turing Test to Tell Computer and Human Apart) is a Human Interactive Proof (HIP) system. The thumb rule of CAPTCHA is that it should be solved easily by a human but not by a bot. In this paper, we introduce a new security primitive based on hard AI problems, namely, a novel family of graphical password systems integrating Captcha technology, which we call *CaRP (Captcha as gRaphical Passwords)*. CaRP is click-based graphical passwords, where a sequence of clicks on an image is used to derive a password. Unlike other click-based graphical passwords, images used in CaRP are Captcha challenges, and a new CaRP image is generated for every login attempt. CaRP offers protection against online dictionary attacks on passwords, which have been for long time a major security threat for various online services. Defense against online dictionary attacks is a more difficult problem than it might appear. It causes denial-of-service attacks (which were exploited to lock highest bidders out in final minutes of eBay auctions [2]) and incurs expensive helpdesk costs for account reactivation.

## II. BACKGROUND AND RELATED WORK

### 2.1 Graphical Passwords:

A large number of graphical password schemes have been proposed. They can be classified into three categories according to the task involved in memorizing and entering passwords: recognition, recall, and cued recall. Each type will be briefly described here. More can be found in a recent review of graphical passwords [1]. A *recognition-based* scheme requires identifying among decoys the visual objects belonging to a password portfolio. A typical scheme is wherein a user selects Different images from a database in creating a password. During authentication, a panel of images is presented for the user click point to her portfolio. This process is repeated several rounds, each round with a different panel. A successful login requires correct selection in each round. The set of images in a panel remains the same between logins, but their locations are permuted.

Cognitive Authentication [6] requires a user to generate a path through a panel of images as follows: starting from the top-left image, moving down if the image is in her portfolio, or right otherwise.

This process is repeated, each time with a different panel. A successful login requires that the cumulative probability that correct answers were not entered by chance exceeds a threshold within a given number of rounds. A *recall-based* scheme requires a user to regenerate the same interaction result without cueing. Draw-A-Secret (DAS) [3] was the first recall-based scheme proposed. A user draws her password on a 2D grid. The system encodes the sequence of grid cells along the drawing path as a user drawn password. Cued Click Points (CCP) [4] is similar to Pass Points but uses one image per click, with the next image selected by a deterministic function. Persuasive Cued Click Points (PCCP) [5] extends CCP by requiring a user to select a point inside a randomly positioned viewport when creating a password, resulting in more randomly distributed click-points in a password. Among the three types, recognition is considered the easiest for human memory whereas pure recall is the hardest [1].

### 2.2 Captcha:

Captcha relies on the gap of capabilities between humans and bots in solving certain hard AI problems. There are two types of visual Captcha: text Captcha and Image-Recognition Captcha (IRC). Machine recognition of non-character objects is far less capable than character recognition. IRCs rely on the difficulty of object identification or classification, possibly combined with the difficulty of object segmentation. Captcha can be circumvented through relay attacks whereby Captcha challenges are relayed to human solvers, whose answers are fed back to the targeted application.

## III. CAPTCHA AS GRAPHICAL PASSWORDS

In a guessing attack, a password guess tested in an unsuccessful trial is determined wrong and excluded from subsequent trials. The number of undetermined password guesses decreases with more trials, leading to a better chance of finding the password.

### 3.1 CaRP: An Overview

In CaRP, a new image is generated for every login attempt, even for the same user. CaRP uses an *alphabet* of visual objects (e.g., alphanumerical characters, similar animals) to generate a CaRP image, which is also a Captcha challenge. A major difference between CaRP images and Captcha images is that all the visual objects in the alphabet should appear in a CaRP image to allow a user to input any password but not necessarily in a Captcha image. Many Captcha schemes can be converted to CaRP schemes, as described in the next subsection. CaRP schemes are clicked-based graphical passwords. According to the memory tasks in memorizing and entering a password, CaRP schemes can be classified into two categories: recognition and a new category, *recognition-recall*, which requires recognizing an image and using the recognized objects as cues to enter a password.

### 3.2 User Authentication with CaRP Schemes

Like other graphical passwords, we assume that CaRP schemes are used with additional protection such as secure channels between clients and the authentication server through Transport Layer Security (TLS). A typical way to apply CaRP schemes in user authentication is as follows. The authentication server *AS* stores a salt $s$ and a hash value $H(\rho, s)$ for each user ID, where $\rho$ is the password of the account and not stored. A CaRP password is a sequence of visual object IDs or clickable-points of visual objects that the user selects. Upon receiving a login request, *AS* generates a CaRP image, records the locations of the objects in the image, and sends the image to the user to click her password.

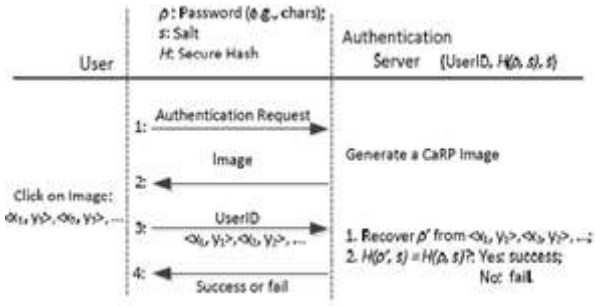


**Fig. 1. Flowchart of basic CaRP authentication**

Authentication succeeds only if the two hash values match. This process is called the *basic CaRP authentication* and shown in Fig. 1. Advanced authentication with CaRP, for example, challenge-response, will be presented in Section V-B. We assume in the following that CaRP is used with the basic CaRP authentication unless explicitly stated otherwise. To recover a password successfully, each user-clicked point must belong to a single object or a clickable-point of an object. Objects in a CaRP image may overlap slightly with neighboring objects to resist segmentation. Users should not click inside an overlapping region to avoid ambiguity in identifying the clicked object. This is not a usability concern in practice since overlapping areas generally take a tiny portion of an object.

## IV. RECOGNITION-BASED CaRP

For this type of CaRP, a password is a sequence of visual objects in the alphabet. Per view of traditional recognition based graphical passwords, recognition-based CaRP seem to have access to an infinite number of different visual objects. We present two recognition-based CaRP schemes and a variation next.

### 4.1 ClickText:

*ClickText* is a recognition-based CaRP scheme built on top of text Captcha. Its alphabet comprises characters without an visually-confusing characters. For example, Letter "O" and digit "0" may cause confusion in CaRP images, and thus one character should be excluded from the alphabet. A ClickText password is a sequence of characters in the alphabet. A ClickText image is generated by the underlying Captcha engine as if a Captcha image were generated except that all the alphabet characters should appear in the image. During generation, each character's location is tracked to produce ground truth for the location of the character in the generated image. The authentication server relies on the ground truth to identify the characters corresponding to user-clicked points.In ClickText images, characters can be arranged randomly.

**Fig. 2. A ClickText image with 33 characters.**

On 2D space. This is different from text Captcha challenges in which characters are typically ordered from left to right in order for users to type them sequentially. Fig. 2 shows a ClickText image with an alphabet of 33 characters. In entering a password, the user clicks on this image the characters in her password, in the same order.

### 4.2. ClickAnimal

Captcha Zoo [7] is a Captcha scheme which uses 3D models of horse and dog to generate 2D animals with different textures, colors, lightings and poses, and arranges them on a cluttered background. A user clicks all the horses in a challenge image to pass the test.

**Fig. 3. Captcha Zoo with horses circled red.**

Fig. 3 shows a sample challenge wherein all the horses are circled red. *ClickAnimal* is a recognition-based CaRP scheme built on top of Captcha Zoo [7], with an alphabet of similar animals such as dog, horse, pig, etc. Its password is a sequence of animal names such as $\rho$ = "Turkey, Cat, Horse, Dog,…." For each animal, one or more 3D models are built. The Captcha generation process is applied to generate ClickAnimal images: 3D models are used to generate 2D animals by applying different views, textures, colors, lightning effects, and optionally distortions. The resulting 2D animals are then arranged on a cluttered background such as grassland. Some animals may be occluded by other animals in the image, but their core parts are not occluded in order for humans to identify each of them.

## V. RECOGNITION-RECALL CaRP

In recognition-recall CaRP, a password is a sequence of some invariant points of objects. An *invariant point* of an object (e.g. letter "A") is a point that has a fixed relative position in different incarnations (e.g., fonts) of the object, and thus can be uniquely identified by humans no matter how the object appears in CaRP images. To enter a password, a user must identify the objects in a CaRP image, and then use the identified objects as cues to

locate and click the invariant points matching her password. Each password point has a tolerance range that a click within the tolerance range is acceptable as the password point. Most people have a click variation of 3 pixels or less [18]. TextPoint, a recognition recall CaRP scheme with an alphabet of characters, is presented next, and followed by a variation for challenge response Authentication.

## VI. TEXTPOINTS

Characters contain invariant points. some invariant points of letter "A", which offers a strong cue to memorize and locate its invariant points. A point is said to be an *internal point* of an object if its distance to the closest boundary of the object exceeds a threshold. A set of internal invariant points of characters is selected to form a set of *clickable points* for TextPoints.

**Fig. 5. Some invariant points (red crosses) of "A".**

For example, if the center of a stroke segment in one character is selected, we should avoid selecting the center of a similar stroke segment in another character. Instead, we should select Fig. 5. A different point from the stroke segment, e.g., a point at one-third length of the stroke segment to an end. This variation in selecting clickable points ensures that a clickable point is context-dependent: a similarly structured point may or may not be a clickable point, depending on the character that the point lies in. Character recognition is required in locating clickable points on a TextPoints image although the clickable points are known for each character. This is a task beyond a bot's Capability. A password is a sequence of clickable points. A character can typically contribute multiple clickable points. Therefore TextPoints has a much larger password space than ClickText.

Image Generation. TextPoints images look identical to ClickText images and are generated in the same way except that the locations of all the clickable points are checked to ensure that none of them is occluded or its tolerance region overlaps another clickable point's. We simply generate another image if the check fails. As such failures occur rarely due to the fact that clickable points are all internal points, the restriction due to the check has a negligible impact on the security of generated images.

Authentication. When creating a password, all clickable points are marked on corresponding characters in a CaRP image for a user to select. During authentication, the user first identifies her chosen characters, and clicks the password points on the right characters. The authentication server maps each user-clicked point on the image to find the closest clickable point. If their distance exceeds a tolerable range, login fails. Otherwise a sequence of clickable points is recovered, and its hash value is computed to compare with the stored value. It is worth comparing potential password points between TextPoints and traditional click-based graphical passwords. In PassPoints, salient points should be avoided since they are readily picked up by adversaries to mount dictionary attacks, but avoiding salient points would increase the burden to remember a password. This conflict does not exist in TextPoints. Clickable points in TextPoints are salient points of their characters and thus help remember

a password, but cannot be exploited by bots since they are both *dynamic* (as compared to static points in traditional graphical password schemes) and *contextual:*

Dynamic: locations of clickable points and their contexts (i.e., characters) vary from one image to another. The clickable points in one image are computationally independent of the clickable points in another image, as we will see in Section VI-B.

Contextual: Whether a similarly structured point is a clickable point or not depends on its context. It is only if within the right context, i.e., at the right location of a right character. i.e., characters, first. By the very nature of Captcha, recognizing characters in a Captcha image is a task beyond computer's capability. Therefore, these salient points of characters cannot be exploited to mount dictionary attacks on TextPoints.

## VI. SECURITY ANALYSIS

### 6.1 Security of Underlying Captcha:

ClickText is much harder to break than its underlying Captcha scheme. Furthermore, characters in a CaRP scheme are arranged two dimensionally, further increasing segmentation difficulty due to one more dimension to segment. As a result, we can reduce distortions in ClickText images for improved usability yet maintain the same security level as the underlying text Captcha. ClickAnimal relies on both object segmentation and multiple-label classification. Its security remains an open question. As a framework of graphical passwords, CaRP does not rely on any specific Captcha scheme. If one Captcha scheme gets broken, a new and more robust Captcha scheme may appear and be used to construct a new CaRP scheme. In the remaining security analysis, we assume that it is intractable for computers to recognize any objects in any challenge image generated by the underlying Captcha of CaRP.

### 6.2 Automatic Online Guessing Attacks:

In automatic online guessing attacks, the trial and error Process is executed automatically whereas dictionaries can be constructed manually. If we ignore negligible probabilities, CaRP with underlying CPA-secure.

### 6.3 Human Guessing Attacks

In human guessing attacks, humans are used to enter passwords in the trial and error process. Humans are much slower than computers in mounting guessing attacks. For 8-character passwords, the theoretical password space is $338 \approx 240$ for ClickText with an alphabet of 33 characters, $108 \approx 226$ for ClickAnimal with an alphabet of 10 animals, and $10 \times 467 \approx 242$.

### 6.4 Relay Attacks:

Relay attacks may be executed in several ways. Captcha challenges can be relayed to a high-volume Website hacked or controlled by adversaries to have human surfers solve the challenges in order to continue surfing the Website, or relayed to sweatshops where humans are hired to solve Captcha challenges for small payments.

### 6.5 Shoulder-Surfing Attacks

Shoulder-surfing attacks are a threat when graphical passwords are entered in a public place such as bank ATM machines. CaRP is not robust to shoulder-surfing attacks by itself. However, combined with the following dual-view technology, CaRP can thwart shoulder-surfing attacks.

### 6.6 Others

CaRP is not bulletproof to all possible attacks. CaRP is vulnerable if a client is compromised such that both the image and user-clicked points can be captured. Like many other graphical passwords such as CCP and PCCP, CaRP schemes using the basic CaRP authentication are vulnerable to phishing since user-clicked points are sent to the authentication server.

## VII. CONCLUSION

We have proposed CaRP, a new security primitive relying on unsolved hard AI problems. CaRP is both a Captcha and a graphical password scheme. Various CAPTCHA alternatives are continuously emerging, and this race will continue as more advanced bots emerge. However, the basic idea of CAPTCHAs is to tell humans and machines apart, and this concept is still worth to be discovered for several reasons. CAPTCHAs today are making use of AI-hard problems such as cognitive skills of human being for preventing bots and thus ensuring the security of web applications. Future trends in CAPTCHA techniques henceforth need to encourage the application of AI-hard problems for efficient prevention of bots. The project will mainly manage the security aspects of the Application. ). CaRP is click-based graphical passwords, where a sequence of clicks on an image is used to derive a password. Images used in CaRP are Captcha challenges, and a new CaRP image is generated for every login attempt.

## REFERENCES

[1] R. Biddle, S. Chiasson, and P. C. van Oorschot, "Graphical passwords: Learning from the first twelve years," *ACM Comput. Surveys*, vol. 44, no. 4, 2012.

[2] T. Wolverton. (2002, Mar. 26). *Hackers Attack eBay Account* .[Online]. Available: http://www.zdnet.co.uk/news/networking/ 2002/03/26/hackers-attack-ebay-accounts-2107350

[3] I. Jermyn, A. Mayer, F. Monrose, M. Reiter, and A. Rubin, "The design and analysis of graphical passwords," in *Proc. 8th USENIX Security Symp.*, 1999, pp. 1–15.

[4] S. Chiasson, P. C. van Oorschot, and R. Biddle, "Graphical password authentication using cued click points," in *Proc. ESORICS*, 2007, pp. 359–374.

[5] S. Chiasson, A. Forget, R. Biddle, and P. C. van Oorschot, "Influencing users towards better passwords: Persuasive cued click-points," in *Proc. Brit. HCI Group Annu. Conf. People Compute. Culture, Creativity, Interaction*, vol. 1. 2008, pp. 121–130.

[6] D. Weinshall, "Cognitive authentication schemes safe against spyware," in *Proc. IEEE Symp. Security Privacy*, May 2006, pp. 300–306.

[7] R. Lin, S.-Y. Huang, G.B.Bell, and Y.-K. Lee, "A new CAPTCHA interface design for mobile devices," in *Proc. 12th Austral. User Inter. Conf.*, 2011, pp. 3–8.

[8] Captcha as Graphical Passwords—A New Security

Primitive Based on Hard AI Problems ,Bin B. Zhu, Jeff Yan, Guanbo Bao, Maowei Yang, and Ning Xu, IEEE transactions on information forensics and security, VOL. 9, NO. 6, JUNE 2014