ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)
Vol. 4, Issue 2 (Apr. - Jun. 2016)
International Journal of Advanced Research in Computer Science & Technology (IJARCST 2016)

Captcha As Graphical Passwords

[I]Giridhar C, [II]Kurian George, [III]Ambikadevi Amma T
[I]M Tech Student, Dept. of CS, J C E T Lakkidi.
[II]Assistant Professor, Dept. of CS, J C ET Lakkidi.
[III]Professor, Head Of Dept. of CS, J C E T Lakkidi.

Abstract
Many security primitives are based on hard mathematical problems. Now a days there are internet bots that are capable of possessing threat to the security mechanisms imposed. The CaRP(Captcha As gRaphical Password) is a novel approach to get away with the security threat imposed by the internet bots. The CaRP is both captcha as well as graphical password scheme. CaRP addresses a number of security problems such as online guessing attacks, sql injection, and relay attacks etc.The CaRP is click based graphical scheme. Where entering the password is by clicking the captcha image provided. On each trial a different captcha image will be provided, which will cut down the possibility of being affected by the attacks. Notably, a CaRP password can be found only probabilistically by automatic online guessing attacks even if the password is in the search set. The CaRP has its implementations in internet banking, digital wallets, online accounts etc. The use of CaRP does not provide much user friendliness since the user has to spend some time in entering the password. Comparing with the current security methods the CaRP is much better than other methods. CaRP offers reasonable security and usability and appears to fit well with some practical applications for improving online security.

Introduction

The most common computer authentication method is for a user to submit a user name and a text password. The vulnerabilities of this method have been well known. One of the main problems is the difficulty of remembering passwords. Studies have shown that users tend to pick short passwords or passwords that are easy to remember [2]. Unfortunately, these passwords can also be easily guessed or broken. According to a recent Computerworld news article, the security team at a large company ran a network password cracker and within 30 seconds, they identified about 80% of the passwords [3]. On the other hand, passwords that are hard to guess or break are often hard to remember. Studies showed that since user can only remember a limited number of passwords, they tend to write them down or will use the same passwords for different accounts [4, 5]. To address the problems with traditional username password authentication, graphical passwords is developed.

Graphical password schemes have been proposed as a possible alternative to text-based schemes, motivated partially by the fact that humans can remember pictures better than text; psychological studies supports such assumption [8]. Pictures are generally easier to be remembered or recognized than text. In addition, if the number of possible pictures is sufficiently large, the possible password space of a graphical password scheme may exceed that of text based schemes and thus presumably offer better resistance to dictionary attacks. Because of these (presumed) advantages, there is a growing interest in graphical password. In addition to workstation and web log-in applications, graphical passwords have also been applied to ATM machines and mobile devices.

A graphical password is easier than a text-based password for most people to remember. Suppose an 8-character password is necessary to gain entry into a particular computer network. Instead of w8KiJ72c, for example, a user might select images of the earth (from among a screen full of real and fictitious planets), the country of France (from a map of the world), the city of Nice (from a map of France), a white stucco house with arched doorways and red tiles on the roof, a green plastic cooler with a white lid, a package of Gouda cheese, a bottle of grape juice, and a pink paper cup with little green stars around its upper edge and three red bands around the middle. Graphical passwords may offer better security than text-based passwords because many people, in an attempt to memorize text-based passwords, use plain words (rather than the recommended jumble of characters). A dictionary search can often hit on a password and allow a hacker to gain entry into a system in seconds. But if a series of selectable images is used on successive screen pages, and if there are many images on each page, a hacker must try every possible combination at random.

Related Work

a). Graphical Passwords

A lot of graphical password schemes have been proposed. They are classified into three categories according to the task involved in memorizing and entering passwords: identification, recall, and cued recall. Every type will be briefly described as follows. Evenmore can be found in a recent review of graphical passwords [1].

A recognition-based scheme requires identifying amongst decoys the pctorial objects belonging to a password portfolio. A typical scheme is Passfaces [2] where a user selects a portfolio of faces from a database in creating a password. During the time of authentication, a panel of candidate faces is presented for the user to select the face belonging to his collection. This process is repeated many rounds, every round with a different panel. A successful login requires correct selection in every round. The set of images in a panel remains the same between logins, but their locations are changed. Story [3] is similar to Passfaces but the images in the collection are ordered, and a user must identify his collection images in the accurate order. Cognitive Authentication [12] requires a user to generate a path through a panel of images as follows: starting through the topmost-left image, moving down if the image is in his collection, or right if not. The user identifies amongst decoys the column or row label that the path ends.

This process is repeated every time with a different panel. For a login to be successful 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 his password on a 2 dimentional grid. The system encodes the series of grid cells along the drawing path as a user-drawn password. Pass-Go [4] improves DAS's usability by encoding the grid intersection points rather than the grid cells. BDAS [3] adds background images to DAS to encourage users

ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

to create more complex passwords.
In a cued-recall scheme, an external cue is provided to help memorize and enter a password. PassPoints [5] is a widely studied click-based cued-recall scheme where a user clicks a series of points anywhere on an image in creating a password, and repeated clicks the same series during the time of authentication. Cued Click Points (CCP) [14] is exactly similar to PassPoints but it uses one image for every click, and successive image selected by using deterministic function. Persuasive Cued Click Points (PCCP) [11] 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.

b). Captcha

Captcha depends on the gap of capabilities between humans and bots in solving certain hard Artificial Intelligence problems. Mainly two types of visual Captcha: text Captcha and Image-Recognition Captcha (IRC). The former depends on character recognition while the latter relies on recognition of non-character objects. Security of text Captchas has been broadly studied [6]–[10]. The following principle has been established: text Captcha should rely on the difficulty of character segmentation and which is computationally expensive and combinatorially hard [5].
Machine recognition of non-character objects is far less capable than character recognition. IRCs depend on the difficulty of object identification or classification, possibly combined with the difficulty of object segmentation. A user is asked to identify all the cats from a panel of 12 images of cats as well as dogs. Security of IRCs is also been studied and found to be susceptible to machine-learning attacks [4]. IRCs based on binary object classification or identification of one concrete type of objects are likely insecure [15]. Multi-label classification problems are considered much harder than binary classification problems.
Captcha may be circumvented during relay attacks whereby Captcha challenges are depended to human solvers, whose answers are fed back to the targeted application.

c). Captcha in Authentication

It was introduced in [14] to use both Captcha and password in a user authentication protocol, which we call Captchabased Password Authentication (CbPA) protocol, to counter online dictionary attacks. The CbPA-protocol in [14] needs solving of a Captcha challenge after giving a valid pair of user ID and password unless a valid browser cookie is acknowledged. For an unsound pair of user ID and password, the user has a certain probability to solve a Captcha challenge before saying no to the access. A modified CbPA-protocol is proposed in [15] by storing cookies only on user-trusted machines and applying a Captcha challenge only when the number of failed login attempts for the account has exceeded a threshold. It is further improved in [6] by applying a small threshold for failed login attempts from unknown machines but a large threshold for failed attempts from known machines with a previous successful login within a given time frame.Captcha was also used with recognition-based graphical passwords to address spyware [8, 9], wherein a text Captcha is displayed below each image; a user locates her own pass-images from decoy images, and enters the characters at specific locations of the Captcha below each pass-image as her password during authentication. These specific locations were selected for each pass-image during password creation as a part of the password.

d) Other Related Work

Captcha is used to protect sensitive user inputs on an untrusted client [15]. This scheme protects the communication channel between user and Web server from keyloggers and spyware, while CaRP is a family of graphical password schemes for user authentication. The paper [15] did not introduce the notion of CaRP or explore its rich properties and the design space of a variety of CaRP instantiations.

Captcha as Graphical Passwords

In CaRP, a new image is generated for every login attempt, even for the same user. CaRP uses an alphabet of visual objects 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. 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. Recognition-recall combines the tasks of both recognition and cued-recall, and retains both the recognition-based advantage of being easy for human memory and the cued-recall advantage of a large password space. In principle, any visual Captcha scheme relying on recognizing two or more predefined types of objects can be converted to a CaRP. All text Captcha schemes and most IRCs meet this requirement. Those IRCs that rely on recognizing a single predefined type of objects can also be converted to CaRPs in general by adding more types of objects.
The flow chart of the basic CaRP authentication is given below in Fig:1. 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(ρ, s) for each user ID, where ρ 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.

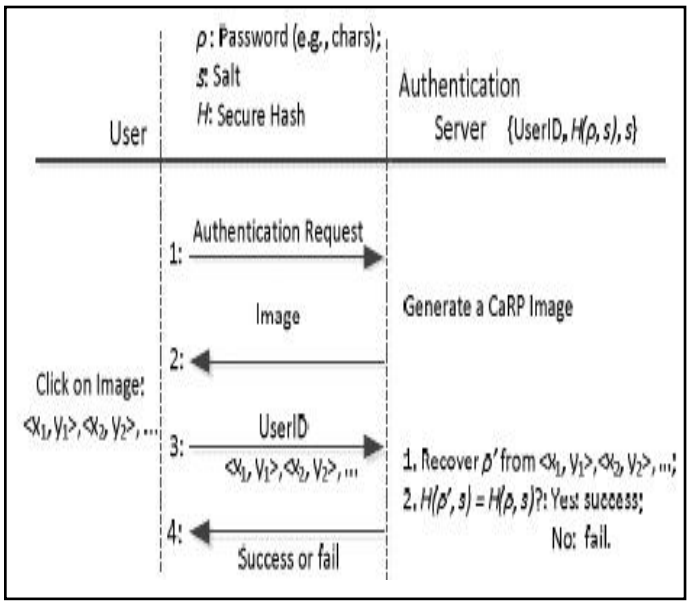

Fig 1: Flow of Basic CaRP Authentication

The coordinates of the clicked points are recorded and sent to AS along with the user ID. AS maps the received coordinates onto the CaRP image, and recovers a sequence of visual object IDs or clickable points of visual objects, ρ', that the user clicked on the image. Then AS retrieves salt s of the account, calculates the hash value of ρ' with the salt, and compares the result with the hash value stored for the account. Authentication succeeds only if the two hash values match. This process is called the basic CaRP authentication and shown in Fig. 1.

Empirical Evaluations

a) Implementation

The technology used for the implementation of the CaRP was Java, with the help of Eclipse IDE. The password size was set between 8 - 30 characters, that is the minimum password size is of 8 character and the maximum is of 30 characters. The password characters can include letters, alphanumeric characters and special characters. The special characters were chosen to balance security and users stong dislike for using non-alphanumeric charcters in the passwords. Each characters was randomly rotated from -30 to 30 and scaled from 60% to 120 %. Neighboring characters Could overlap up to 3 pixels. Wraping effects were set to the light level. Each image was set to 400 by 400 pixels. On each instance a separate captcha is provided, which will be independent of the previous one. The bulk of the computations was done at the server side, the server used was Apache Tomcat server which is the default server in the eclipse IDE. The HTML and CSS was used for the designing of the web pages. Each password met the following minimum complexity requirements . A password must at least contain one letter, one digit, one alphanumerical character. Each password was verified after immediate creation. P+C was used to simulate CaRP that is password combined with the text captcha.

A user's login time in each trial was recorded by the server. We define the login time as the duration from the time when the server received a login request to the time when the server gave its response to the login request, which includes the time to enter user ID and password, to generate a CaRP image, and to communicate between the server and a participant's browser.

For Text and P + C, a participant was asked to enter a password. If successful, the server recorded the time as the login time for Text, and then generated a Captcha challenge and sent to the user to solve. If the participant failed with the challenge, another challenge was generated and used. This process was repeated until the server received a correct answer to a challenge. Then the server recorded the time as the login time for P + C, which included the time that the participant failed to solve a challenge.

b) Usability

The CaRP can be implemented in any system without the addition of any other software or hardware , this shows the usability of the CaRP approach. Moreover the users are able to memorize their passwords more easily. Increasing alphabet size produces a larger password space, and thus is more secure, but also leads to more complex CaRP images. When the complexity of CaRP images gets beyond a certain point, humans may need a significant amount of time to recognize the characters in a CaRP image and may get frustrated.. The optimal alphabet size for a CaRP scheme such as Click Text remains an open question. It is possible to use a fixed subset of the alphabet to generate CaRP images for a user if the server receives her user ID before sending an image. In this case, the authentication server allows a user to create her password from the full alphabet. Once the password is created, the server finds a suitable subset of a reasonable size, which contains all the symbols in the password. The server stores the subset or its index for the account, and retrieves it later when the account attempts to log in to generate a CaRP image. This scheme is suitable when the alphabet must be large while some people would log in on small-screen devices for which an image using the full alphabet would be too complex to quickly identify the objects in the image.

c) Security

The CaRP provides better security against the bots attacks. The bots are the software programs that are capable of cracking our passwords. The CaRP requires some human intelligence while typing our password since we are making use of the captchas. This is where the bots attacks can be countered the bots does not has their own intelligence, only a human is capable of solving the captcha challenge. Thus the CaRP provides better security .

Conclusion

CaRP is both a Captcha and a graphical password scheme. The notion of CaRP introduces a new family of graphical passwords, which adopts a new approach to counter online guessing attacks: a new CaRP image, which is also a Captcha challenge, is used for every login attempt to make trials of an online guessing attack computationally independent of each other. A password of CaRP can be found only probabilistically by automatic online guessing attacks including brute-force attacks, a desired security property that other graphical password schemes lack. Hotspots in CaRP images can no longer be exploited to mount automatic online guessing attacks, an inherent vulnerability in many graphical password systems. CaRP forces adversaries to resort to significantly less efficient and much more costly human-based attacks. In addition to offering protection from online guessing attacks, CaRP is also resistant to Captcha relay attacks, and, if combined with dual-view technologies, shoulder-surfing attacks. CaRP can also help reduce spam emails sent from a Web email service.

References

[1]. *Zhu, B.B.; Yan, J.; Guanbo Bao; Maowei Yang; Ning Xu, "Captcha as Graphical Passwords—A New Security Primitive Based on Hard AI Problems," in Information Forensics and Security, IEEE Transactions on , vol.9, no.6, pp.891-904, June 2014*

[2]. *P. C. van Oorschot and J. Thorpe, "On predictive models and userdrawn graphical passwords," ACM Trans. Inf. Syst. Security, vol. 10, no. 4, pp. 1–33, 2008.*

[3]. *K. Golofit, "Click passwords under investigation," in Proc. ESORICS, 2007, pp. 343– 358. [8] A. E. Dirik, N. Memon, and J.-C. Birget,*

[4]. *"Modeling user choice in the passpoints graphical password scheme," in Proc. Symp. Usable Privacy Security, 2007, pp. 20–28.*

[5]. *J. Thorpe and P. C. van Oorschot, "Human-seeded attacks and exploiting hot spots in graphical passwords," in Proc. USENIX Security, 2007, pp. 103–118.*

[6]. *P. C. van Oorschot, A. Salehi-Abari, and J. Thorpe, "Purely automated attacks on passpoints-style graphical passwords," IEEE Trans. Inf.Forensics Security, vol. 5, no. 3, pp. 393–405, Sep. 2010.*

ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

[7]. P. C. van Oorschot and J. Thorpe, "Exploiting predictability in clickbased graphical passwords," J. Comput. Security, vol. 19, no. 4, pp. 669–702, 2011.

[8]. M. Alsaleh, M. Mannan, and P. C. van Oorschot, "Revisiting defenses against large-scale online password guessing attacks," IEEE Trans. Dependable Secure Comput., vol. 9, no. 1, pp. 128–141, Jan./Feb. 2012.

[9]. L. von Ahn, M. Blum, N. J. Hopper, and J. Langford, "CAPTCHA: Using hard AI problems for security," in Proc. Eurocrypt, 2003, pp. 294–311.

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

[11]. 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 Comput., Culture, Creativity,Interaction, vol. 1. 2008, pp. 121– 130.

[12]. [11] D. Davis, F. Monrose, and M. Reiter, "On user choice in graphical password schemes," in Proc. USENIX Security, 2004, pp. 1–11.

[13]. R. Dhamija and A. Perrig, "Déjà Vu: A user study using images for authentication," in Proc. 9th USENIX Security, 2000, pp. 1–4.

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

[15]. P. Dunphy and J. Yan, "Do background images improve 'Draw a Secret' graphical passwords," in Proc. ACM CCS, 2007, pp. 1–12.

[16]. P. Golle, "Machine learning attacks against the Asirra CAPTCHA," in Proc. ACM CCS, 2008, pp. 535–542.

[17]. B. B. Zhu et al., "Attacks and design of image recognition CAPTCHAs," in Proc. ACM CCS, 2010, pp. 187–200.