International Journal On Engineering Technology and Sciences – IJETS™
ISSN(P): 2349-3968, ISSN (O): 2349-3976
Volume I, Issue IV, August - 2014

Captcha as Graphical Password- Based on Hard AI Problems

S.Navaneethakrishnan, P.Kumar

Student, Assistant professor (CSE)

Nandha College Of Technology,Erode

snkrish1990@gmail.com csekumar@gmail.com

ABSTRACT: ***Many security primitives are based on the hard mathematical problems. Using this hard AI problems for security is emerging as an exciting new paradigm, but it has been under explored. This paper, we present a new security primitive based on hard AI problems that is a novel family of graphical password systems built on top of Captcha technology, which we called Captcha as a graphical passwords (CaRP). CaRP is both a Captcha and the graphical password scheme. CaRP was addresses a number of security problems that all together, such as online guessing attacks like relay attacks. if we combined with dual-view technologies, shoulder-surfing attacks. A CaRP password can be found only probabilistically by automatic online guessing attacks even if the password is in the search sets. CaRP also offers that a novel approach to address the well-known image hot spot problem in a popular graphical password systems like Pass Points, that often leads to weak password choices. CaRP is not a panacea, but its offer reasonable security and usability and appears to fit well with some of the practical applications for improving online security.***

Index Terms – CaRP, Captcha, dictionary attack, Graphical password, hotspots, password guessing attack, security primitive.

I.INTRODUCTION

A fundamental task in security is to create crypto- graphic primitives based on hard mathematical problems that are computationally intractable. For example, the problem of integer factorization is fundamental to the RSA public-key cryptosystemand the Rabin encryption. The discrete logarithm problem is fundamental to the ElGamal encryption, the Diffie- Hellman key exchange, the Digital Signature Algorithm, the elliptic curve cryptography and soon.Using hard AI (Artificial Intelligence) problems for security, initially proposed in [7], is an exciting new paradigm. Under this paradigm, the most notable primitive invented is Captcha, which distinguishes human users from computers by presenting a challenge, i.e., a puzzle, beyondthe capability of computers but easy for humans. Captcha is now a standard Internet security technique to protect online email and other services from being abused by bots. However, this new paradigm has achieved just a limited success as compared with the cryptographic primitives based on hard math problems and their wide applications. Is it possible to create any new security primitive based on hard AI problems? This is a challenging and interesting open problem.

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 RaphicalPasswords). CaRP is click-based graphicalpasswords,wherea sequenceofclicks

onanimageis 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. The notion of CaRP is simple but generic. CaRP can have multiple instantiations. In theory, any Captcha scheme relying on multiple-object classification can be converted to a CaRP scheme. We present exemplary CaRPs built on both text Captcha and image-recognition Captcha. One of them is a text CaRP wherein a password is a sequence of characters like a text password, but entered by clicking the right character sequence on CaRP images. CaRP offers protection against online dictionary attacks on passwords, which have been for long time a major security threat for various online services. This threat is widespread and considered as a top cyber security risk [13]. Defense against online dictionary attacks is a more subtle problem than it might appear. Intuitive countermeasures such as throttling logon attempts do not work well for two reasons:

1) It causes denial-of-service attacks (which were exploited to lock highest bidders out in final minutes of eBay auctions [12]) and incurs expensive helpdesk costs for account reactivation.

2) It is vulnerable to global password attacks [14] whereby adversaries intend to break into any account rather than a specific one, and thus try each password candidate on multiple accounts and ensure that the number of trials on each account is below the threshold to avoid triggering account lockout.

II. BACKGROUND AND RELATED WORK

A.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 identifyingamong decoys the visual objects belonging to a password portfolio. A typical scheme is Pass faces [2] wherein a user selects a portfolio of faces from a database in creating a password. During authentication, a panel of candidate faces is presented for the user to select the face belonging 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. Story [20] is similar to Pass faces but the images in the portfolio are ordered, and a user must identify her portfolio images in the correct order. Déjà Vu [21] is also similar but uses a large set of computer- generated "random-art" images. Cognitive Authentication [22] 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.

The user identifies among decoys the row or column label that the path ends.

B. 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). The former relies on character recognition while the latter relies on recognition of non-character objects. Security of text Captchas has been extensively studied [26]–[30]. The following principle has been established: text Captcha should rely on the difficulty of character segmentation, which is computationally expensive and combinatorially

hard [30]. 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.

Asirra [31] relies on binary object classification: a user is asked to identify all the cats from a panel of 12 images of cats and dogs. Security ofIRCs has also been studied. Asirra was found to be susceptible to machine-learning attacks [24]. IRCs based on binary object classification or identification of one concrete type of objects are likely insecure [25]. Multi-label classification problems are considered much harder than binary classification problems. Captcha can be circumvented through relay attacks whereby Captcha challenges are relayed 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 Captcha-based Password Authentication (CbPA) protocol, to counter online dictionary attacks. The CbPA-protocol in [14] requires solving a Captcha challenge after inputting a valid pair of user ID and password unless a valid browser cookie is received. For an invalid pair of user ID and password, the user has a certain probability to solve a

Captcha challenge before being denied access. An improved 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 [16] 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 [40], [41], 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. In the above schemes, Captchais anindependententity,used together with a text or graphical password. On the contrary, a CaRP is both a Captcha and a raphical password scheme, which are intrinsically combined into a single entity.

III. CAPTCHA AS GRAPHICAL PASSWORDS

A New Way to Thwart Guessing Attacks 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 moretrials, leadingto a better chance of finding the password. Mathematically, let S be the set of password guesses before any trial, ρ be the password to find, T denotea trial whereas Tn denote the n-th trial, and $p(T = \rho)$ be the probability that ρ is tested in trial T. LetEn be the set of password guesses tested in trials up to (including) Tn.

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 seems to have access to an infinite number of different visual objects. We present two recognition-based CaRP schemes and a variation next.

International Journal On Engineering Technology and Sciences – IJETS™
ISSN(P): 2349-3968, ISSN (O): 2349-3976
Volume I, Issue IV, August - 2014

A. ClickText

ClickText is a recognition-based CaRP scheme built on top of text Captcha. Its alphabet comprises characters without any 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, e.g., ρ ="AB#9CD87", which is similar to a text password. 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 Click Text image with 33 characters.

C. AnimalGrid

The number of similar animals is much less than the number of available characters. Click Animal has a smaller alphabet, and thus a smaller password space ,than Click Text. CaRP should have a sufficiently-large effective password space to resist human guessing attacks. Animal Grid's password space can be increased by combining it with a grid-based graphical password, with the grid depending on the size of the selected animal. DAS [3] is a candidate but requires drawing on the grid. To be consistent with Click Animal, we change from drawing to clicking: Click-A-Secret (CAS) wherein a user clicks the grid cells in her password. Animal Grid is a combination of Click Animal and CAS.

The number of grid-cells in a grid should be much larger than the alphabet size. Unlike DAS, grids in our CAS are object-dependent, as we will see next. It has the advantage that a correct animal should be clicked in order for the clicked grid-cell(s) on the follow-up grid to be correct. If a wrong animal is clicked, the follow-up grid is wrong. A click on the correctly labeled grid-cell of the wrong grid would likely produce a wrong grid-cell at the authentication server side when the correct grid is used. To enter a password, a Click Animal image is displayed first.

After an animal is selected, an image of n×n grid appears, with the grid-cell size equaling the bounding rectangle of the selected animal. Each grid-cell is labeled to help users identify. Fig. 4 shows a 6×6 grid when the red turkey in the left image of Fig. 4 was selected. A user can select zero to multiple grid-cells matching her password. Therefore a password is a sequence of animals interleaving with grid-cells, e.g., ρ = "Dog, Grid2, Grid 1; Cat, Horse, Grid 3", where Grid1 means the grid-cell indexed as 1, and grid-cells after an animal means that the grid is determined by the bounding rectangle of the animal.

A password must begin with an animal. When a ClickAnimal image appears, the user clicks the animal on the image that matches the first animal in her password. The coordinates of the clicked point are recorded.

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, followed by a variation for challenge- response authentication.

A. TextPoints

Characters contain invariant points. Fig. 5 shows 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.

Scheme	ClickText	Animal Grid	PassPoints	P+C	Text
T (s)	27.22	29.20	21.62	28.24	10.34
σ (s)	17.38	19.24	12.29	12.55	6.08
Max.(s)	65.62	88.51	45.17	50.84	31.25
Min.(s)	10.41	13.46	8.36	13.7	3.58

TABLE I : LOGIN TIME FOR DIFFERENT SCHEMES: AVERAGE (T), SAMPLE STANDARD DEVIATION (σ), MAX. AND MIN.

VI. BALANCE OF SECURITY AND USABILITY

Some configurations of CaRP offer acceptable usability across common device types, e.g. our usability studies used 400 × 400 images, which fit displays of smart phones, iPads, and PCs.

While CaRP may take a similar time to enter a password as other graphical password schemes, it takes a longer time to enter a password than widely used text passwords. We discuss two approaches for balancing CaRP's security and usability.

A.Alphabet Size

Increasing alphabet size produces a larger password space, and thus is more secure, but also leads to more complex CaRPimages. 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

B. Advanced Mechanisms

The CbPA-protocols described in Section II-C require a user to solve a Captcha challenge in addition to inputting a password under certain conditions. For example, the scheme describedin [16]applies a Captchachallengewhenthe number of failed login attempts has reached a threshold for an account. A small threshold is applied for failed login attempts from unknown machines but a large threshold is applied for failed attempts from known machines on which a successful login occurred within a given time frame. This technique can be integrated into CaRP to enhance usability: 1. A regular CaRP image is applied when an account has reached a threshold of failed login attempts.

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

[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] H. Tao and C. Adams, Pass-Go: A proposal to improve the usability of graphical passwords," Int. J. Netw. Security, vol. 7, no. 2, pp. 273–292, 2008.

5.Wiedenbeck, J. Waters, J. C. Birget, A. Brodskiy, and N. Memon, "PassPoints: Design and longitudinal evaluation of a graphical password system," Int. J. HCI, vol.

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

[7] K. Golofit, "Click passwords under investigation," in Proc. ESORICS, 2007, pp. 343–358.

[8] A. E. Dirik, N. Memon, and J.-C.Birget, "Modeling user choice in the passpoints graphical password scheme," in Proc. Symp. Usable Privacy Security, 2007, pp. 20–28.

[9] 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.

[10] 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.

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

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

[13] HP TippingPointDVLabs, Vienna, Austria. (2010). Top Cyber Security Risks Report, SANS Institute and Qualys Research Labs [Online].

[14] P. C. van Oorschot and S. Stubblebine, "On countering online dictionary attacks with login histories and humans-in-the-loop," ACM Trans. Inf. Syst. Security, vol.

[16] 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.

[17] 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.