ISSN:0975-9646

Captcha based Password Authentication - A New Security Scheme

Monika Chilluru[1], B. Ravindra Naick[2], P. Nirupama[3]

[1]*M.Tech Student,* [2]*Assistant Professor,* [3]*Head of the Department*
[1, 2, 3]*Department of Computer Science and Engineering*
[1, 2, 3] *Siddharth Institute of Engineering and Technology*

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

Keywords— Graphical password, password, hotspots, CaRP, Captcha, dictionary attack, password guessing attack, security primitive.

I. INTRODUCTION

FUNDAMENTAL task in security is to make crypto-graphic primitives supported exhausting mathematical issues that area unit computationally intractable, as an example, the matter of number resolving is prime to the RSA public-key cryptosystem and therefore the Rabin cryptography. The separate exponent downside is prime to the ElGamal cryptography, the Diffie-Hellman key exchange, the Digital Signature algorithmic program, the elliptic curve cryptography then on.

Using exhausting AI (Artificial Intelligence) issues for security, ab initio projected in [17], is associate degree exciting new paradigm. Below this paradigm, the foremost notable primitive fictitious is Captcha, that distinguishes human users from computers by presenting a challenge, i.e., a puzzle, beyond the. Capability of computers however simple for humans Captcha is currently a typical web security technique to guard on-line email and alternative services from being abused by bots.

However, this new paradigm has achieved simply a restricted success as compared with the science primitives supported exhausting scientific discipline issues and their wide applications. Is it potential to make any new security primitive supported exhausting AI downsides? This can be a difficult and fascinating open problem. during this paper, we have a tendency to introduce a replacement security primitive supported exhausting AI issues, namely, a completely unique family of graphical countersign systems group action Captcha technology, that we have a tendency to decision CaRP (Captcha as gRaphical Passwords). CaRP is click-based graphical passwords, wherever a sequence of clicks on a picture is employed to derive a countersign. In contrast to alternative click-based graphical passwords, pictures employed in CaRP area unit Captcha challenges, and a replacement CaRP image is generated for each login try.

The notion of CaRP is straightforward however generic. CaRP will have multiple instantiations. In theory, any Captcha theme looking forward to multiple-object classification will be born-again to a CaRP theme. We have a tendency to gift exemplary CaRPs engineered on each text Captcha and image-recognition Captcha. One in all them could be a text CaRP whereby a countersign could be a sequence of characters sort of a text countersign, however entered by clicking the proper character sequence on CaRP Pictures

CaRP offers protection against on-line wordbook attacks on passwords that are for durable a significant security threat for varied on-line services. This threat is widespread and regarded as a high cyber security risk. Defence against on-line wordbook attacks could be a lot of delicate downside than it'd seem. Intuitive countermeasures like suffocation logon makes an attempt don't work well for 2 reasons:

1) It causes denial-of-service attacks (which were exploited to lock highest bidders come in final minutes of eBay auctions) and incurs pricey service prices for account reactivation.
2) It's susceptible to international countersign attacks [14] whereby adversaries will entered any account instead of a selected one, and therefore attempt every countersign candidate on multiple accounts and make sure that the amount of trials on every account is below the edge to avoid triggering account apposition.

CaRP additionally offers protection against relay attacks, associate degree increasing threat to bypass Captchas protection, whereby Captcha challenges area unit relayed to humans to resolve. Koobface [33] was a relay attack to bypass Facebook's Captcha in making new accounts. CaRP is strong to shoulder-surfing attacks if combined with dual view technology. CaRP needs resolution a Captcha challenge in each login. This impact on usability will be

satisfied by adapting the CaRP image's issue level supported the login history of the account and therefore the machine accustomed log in. Typical application eventualities for CaRP include:

1) CaRP will be applied on touch-screen devices whereon writing passwords is cumbersome, esp. for secure web applications like e-banks. Several e-banking systems have applied Captchas in user logins [39]. As an example, ICBC (www.icbc.com.cn), the biggest bank within the world, needs resolution a Captcha challenge for each online login try.
2) CaRP will increase spammer's disbursal and therefore helps scale back spam emails. For associate degree email service supplier that deploys CaRP, a spam larva cannot log into associate degree email account albeit it is aware of the countersign. Instead, human involvement is mandatory to access associate degree account. If CaRP is combined with a policy to throttle the amount of emails sent to new recipients per login session, a spam larva will send solely a restricted range of emails before asking human help for login, resulting in reduced departing spam traffic.

The remaining paper is organized as follows: Related work is presented in Section II. We outline CaRP in Section III, and present a variety of CaRP schemes in Sections IV and V. Security analysis is provided in Section VI. We conclude the paper with Section VII.

II. RELATED WORK

A. Graphical Passwords

A large range of graphical countersign schemes are proposed. They will be classified into 3 classes according to the task concerned in memorizing and getting into passwords: recognition, recall, and cued recall. Every kind is in short described here. A lot of are often found in a very recent review of graphical passwords [1]. A recognition-based theme needs characteristic among decoys the visual objects happiness to a countersign portfolio. A typical theme is Passfaces [2] whereby a user selects a portfolio of faces from a info in making a countersign. During authentication, a panel of the candidate faces is given for the user to pick the face happiness to her portfolio. This process is continual many rounds, every spherical with a unique panel. A palmy login needs correct choice in every round. The set of pictures in a very panel remains identical between logins, however their locations area unit permuted. Story [20] is comparable to Passfaces however the photographs within the portfolio area unit ordered, and a user should determine her portfolio pictures within the correct order. Déjà Vu [21] is additionally similar however uses an oversized set of computer generated "random-art" pictures. Psychological feature Authentication [22] requires a user to get a path through a panel of pictures as follows: ranging 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 trail ends. This method is continual, when with a unique panel. A palmy login needs that the accumulative likelihood that correct answers weren't entered by accident exceeds a threshold among a given range of rounds. A recall-based theme needs a user to regenerate the same interaction result while not cueing. Draw-A-Secret(DAS) [3] was the primary recall-based theme projected. A user draws her countersign on a second grid. The system encodes the sequence of grid cells on the drawing path as a user drawn password. Pass-Go [4] improves DAS's usability by encoding the grid intersection points instead of the grid cells. BDAS [23] adds background pictures to DAS to encourage users to form a lot of advanced passwords. In a cued-recall theme, associate degree external cue is provided to assist memorize and enter a countersign. PassPoints [5] may be a wide studied click-based cued-recall theme whereby a user clicks a sequence of points anyplace on a picture in making a password, and re-clicks identical sequence throughout authentication. Cued Click Points (CCP) [18] is comparable to PassPoints but uses one image per click, with succeeding image designated by a settled perform. Persuasive Cued Click Points (PCCP) [19] extends CCP by requiring a user to pick a point within a willy-nilly positioned viewport once making a password, leading to a lot of willy-nilly distributed click-points in a countersign. Among the 3 varieties, recognition is taken into account the best for human memory whereas pure recall is that the hardest [1]. Recognition is often the weakest in resisting approximation attacks. Several projected recognition-based schemes much have a countersign house within the vary of 213 to 216 passwords [1]. A study [6] according that a major portion of passwords of DAS and Pass-Go [4] were with success broken with approximation attacks victimisation dictionaries of 231 to 241 entries, as compared to the full countersign house of 258 entries. Pictures contain hotspots [7], [8], i.e., spots seemingly designated in making passwords.

Hotspots were exploited to mount palmy approximation attacks on PassPoints [8]–[11]: a major portion of passwords were broken with dictionaries of 226 to 235 entries, as compared to the full house of 243 passwords.

B. Captcha

Captcha depends on the gap of capabilities between humans and bots in determination sure arduous AI issues. There are a unit 2 types of visual Captcha: text Captcha and Image- recognition Captcha (IRC). The previous depends on character recognition while the latter depends on recognition of non-character objects. Security of text Captchas has been extensively studied [26]–[30]. The subsequent principle has been established: text Captcha ought to suppose the problem of character segmentation, which is computationally costly and combinatorial hard [30]. Machine recognition of non-character objects is way less capable than character recognition. IRCs suppose the problem of object identification or classification, probably combined with the problem of object segmentation. Asirra [31] depends on binary object classification: a user is asked to spot all the cats from a panel of twelve pictures of cats and dogs. Security of IRCs has conjointly been studied. Asirra was found to be inclined to machine-learning attacks [24]. IRCs supported binary object classification or identification of 1 concrete kind of objects are possible

insecure [25]. Multi-label classification issues area unit considered a lot of tougher than binary classification issues. Captcha may be circumvented through relay attacks whereby Captcha challenges area unit relayed to human solvers, whose answers area unit fed back to the targeted application.

C. Captcha in Authentication

It was introduced in [14] to use each Captcha and secret in a user authentication protocol, that we tend to decision Captcha-based Password Authentication (CbPA) protocol, to counter on-line dictionary attacks. The CbPA-protocol in [14] needs resolution a Captcha challenge once inputting a legitimate combine of user ID and password unless a legitimate browser cookie is received. For an invalid combine of user ID and secret, the user features a sure probability to resolve a Captcha challenge before being denied access. AN improved CbPA-protocol is projected in [15] by storing cookies solely on user-trusted machines and applying a Captcha challenge only the quantity of unsuccessful login attempts for the account has exceeded a threshold. It's additional improved in [16] by applying little threshold for unsuccessful login makes an attempt from unknown machines however an oversized threshold for unsuccessful makes an attempt from known machines with a previous successful login inside a given timeframe. Captcha was additionally used with recognition-based graphical passwords to deal with spyware [40], [41], whereby a text Captcha is displayed below every image; a user locates her own pass-images from decoy pictures, and enters the characters at specific locations of the Captcha below every pass-image as her secret throughout authentication. These specific locations were chosen for every pass-image throughout secret creation as a neighborhood of the secret. In the on top of schemes, Captcha is AN freelance entity, used together with a text or graphical secret. On the contrary, a CaRP is each a Captcha and a graphical secret theme, which are as such combined into one entity.

D. Other Related Work

Captcha is used to protect sensitive user inputs on an untrusted client [35]. 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 [35] did not introduce the notion of CaRP or explore its rich properties and the design space of a variety of CaRP instantiations.

III. CAPTCHA AS GRAPHICAL PASSWORD

A. A New Way to Thwart Guessing Attacks

In a shot attack, a positive identification guess tested in associate degree unsuccessful trial is set wrong and excluded from ensuant trials. The quantity of undetermined positive identification guesses decreases with additional trials, resulting in an improved likelihood of finding the positive identification. Mathematically, let S be the set of positive identification guesses before any trial, ρ be the positive identification to seek out, T denote a trial whereas TN denote the n-th trial, and $p(T = \rho)$ be the likelihood that ρ is tested in trial T. Let linear unit be the set of positive identification guesses tested in trails up to (including) TN. The password guess to be tested in n-th trial TN is from set S\En−1, i.e., the relative complement of En-1 in S. If $\rho \in S$ then we have

$$p(T=\rho|T1=\rho,..,Tn-1=\rho)>p(T=\rho),(1)$$

and

$$En\rightarrow S$$

$$p(T=\rho|T1=\rho,...,Tn-1=\rho)\rightarrow 1withn\rightarrow|S|,(2)$$

where |S| denotes the cardinality of S. From Eq. (2), the password is usually found at intervals |S| trials if it's in S; otherwise S is exhausted once |S| trials. every trial determines if the tested positive identification guess is that the actual positive identification or not, and the trial's result's settled. To counter shot attacks, ancient approaches in designing graphical passwords aim at increasing the effective password area to create positive identifications more durable to guess and therefore require additional trials. In spite of however secure a graphical positive identification scheme is, the positive identification will continuously be found by a brute force attack. During this paper, we tend to distinguish two sorts of shot attacks: automatic shot attacks apply associate degree automatic trial and error method however S is manually created whereas human shot attacks apply a manual trial and error method. CaRP adopts a totally completely different approach to counter automatic shot attacks. It aims at realizing the subsequent equation:

$$p(T=\rho|T1,...,Tn-1)=p(T=\rho),\forall n,(3)$$

in associate degree automatic shot attack. Eq. (3) means every trial is computationally freelance of alternative trials. Specifically, no matter what number trials dead antecedently, the chance of finding the positive identification with in the current trail continuously remains the same. That is, a positive identification in S is found solely probabilistically by automatic shot (including brute-force) attacks, in distinction to existing graphical positive identification schemes where a positive identification is found at intervals a set variety of trials. How to win the goal? If a replacement image is employed for every trial, and pictures of various trials area unit freelance of every other, then Eq. (3) holds. freelance pictures among completely different login makes an attempt should contain invariant info so the authentication server will verify claimants. By examining the ecosystem of user authentication, we tend to detected that human users enter passwords throughout authentication, whereas the trial and error method in the shot attacks is dead mechanically. The capability gap between humans and machines is exploited to get pictures so they're computationally independent yet retain invariants that solely humans will establish, and therefore use as passwords. The invariants among pictures must be defiant to machines to thwart automatic shot attacks. This demand is that the same as that of a perfect Captcha [25], resulting in creation of CaRP, a replacement family of graphical passwords strong to on-line shot attacks.

B. CaRP: An Overview

In CaRP, a brand new image is generated for each login try, even for constant user. CaRP uses AN alphabet of visual objects (e.g., alphanumerical characters, similar animals) to generate a CaRP image, that is additionally a Captcha challenge. A major distinction between CaRP pictures and Captcha pictures is that each one the visual objects within the alphabet ought to seem in a CaRP image to permit a user to input any watchword however not essentially in a very Captcha image. several Captcha schemes can be born-again to CaRP schemes, as delineate within the next subsection. CaRP schemes are clicked-based graphical passwords. According to the memory tasks in memorizing and coming into a password, CaRP schemes is classified into 2 categories: recognition and a brand new class, recognition-recall, which requires recognizing a picture and exploitation the recognized objects as cues to enter a watchword. Recognition-recall combines the tasks of each recognition and cued-recall, and retains each the recognition-based advantage of being straightforward for human memory and therefore the cued-recall advantage of an oversized password house. Exemplary CaRP schemes of every kind can be given later.

C. Converting Captcha to CaRP

In principle, any visual Captcha theme looking forward to recognizing two or additional predefined kinds of objects are often regenerate to a CaRP. All text Captcha schemes and most IRCs meet this requirement. Those IRCs that think about recognizing one predefined sort of objects may also be regenerate to CaRPs in general by adding additional kinds of objects. In follow, conversion of a particular Captcha theme to a CaRP theme typically needs a case by case study, so as to make sure both security and value. we'll gift in Sections IV and V many CaRPs designed on high of text and image-recognition Captcha schemes. Some IRCs think about characteristic objects whose varieties aren't predefined. A typical example is Cortcha [25] that depends on context-based visual perception whereby the item to be recognized are often of any sort. These IRCs cannot be regenerate into CaRP since a collection of pre-defined object varieties is important for constructing a Arcanum.

D. User Authentication With CaRP Schemes

Like different graphical passwords, we have a tendency to assume that CaRP schemes area unit used with extra protection like secure channels between purchasers and also the authentication server through Transport Layer Security(TSL). A typical thanks o applies CaRP schemes in user authentication are as follows. The authentication server AS stores a salt s and a hash worth H(ρ, s) for each user ID, wherever ρ is that the watch-word of the account and the not stored. A CaRP watchword could be 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 within the image, and sends the image to the user to click her watchword. The coordinates of the clicked points area unit recorded and sent to AS on 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 worth of p with the salt, and compares the result with the hash worth hold on for the account. Authentication succeeds as long as the 2 hash values match. This method is called the fundamental CaRP authentication and shown in Fig. 1. Advanced authentication with CaRP, as an example, challenge-response, are going to be conferred in Section V-B. We assume within the following that CaRP is employed with the fundamental CaRP authentication unless expressly expressed otherwise. To recover a watchword with success, every user-clicked purpose must belong to one object or a clickable- point of associate degree object. Objects in an exceedingly CaRP image might overlap slightly with neighboring objects to resist segmentation. Users shouldn't click within associate degree overlapping region to avoid ambiguity in identifying the clicked object. this can be not a usability concern in practice since overlapping areas usually take a small portion of associate degree object

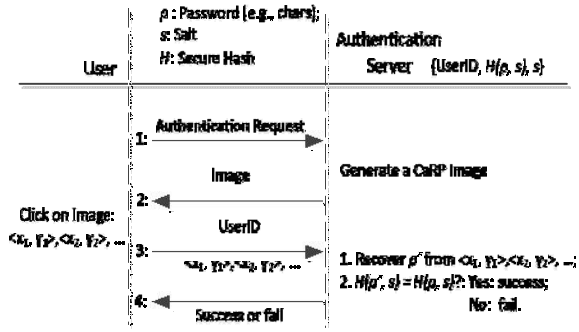

Fig. 1. Flowchart of basic CaRP authentication.

IV. RECOGNITION-BASED CARP

For this kind of CaRP, a countersign could be a sequence of visual objects within the alphabet. Per read of ancient recognition-based graphical passwords, recognition-based CaRP looks to have access to AN infinite range of various visual objects. we have a tendency to gift 2 recognition-based CaRP schemes and a variation next.

A. ClickText

ClickText may be a recognition-based CaRP theme engineered on prime of text Captcha. Its alphabet contains characters with none visually-confusing characters. as an example, Letter "O" and digit "0" might cause confusion in CaRP pictures, and so one character ought to be excluded from the alphabet. A ClickText password may be a sequence of characters within the alphabet, e.g., ρ ="AB#9CD87", that is analogous to a text Arcanum. A ClickText image is generated by the underlying Captcha engine as if a Captcha image were generated except that everyone the alphabet characters ought to seem within the image. During generation, every character's location is half-track to provide ground truth for the situation of the character within the generated image. The authentication server depends on the bottom truth to identify the characters adore user-clicked points. In ClickText pictures, characters are organized willy-nilly on second area. this can

be totally different from text Captcha challenges in which characters area unit generally ordered from left to right in order for users to kind them consecutive. Fig. 2 shows a ClickText image with Associate in Nursing alphabet of thirty three characters. In coming into a password, the user clicks on this image the characters in her password, within the same order, as an example "A", "B", "#", "9", "C", "D", "8", and so "7" for Arcanum ρ = "AB#9CD87".

Fig. 2. A ClickText image with 33 characters.

Fig. 3. Captcha Zoo with horses circled red.

Fig. 4. A ClickAnimal image (left) and 6 × 6 grid (right) determined by red turkey's bounding rectangle.

B. ClickAnimal

Captcha menagerie [32] may be a Captcha theme that uses 3D models of horse and dog to come up with 2nd animals with different textures, colors, lightings and poses, and arranges them on a littered background. A user clicks all the horses in a challenge image to pass the take a look at. Fig. three shows a sample challenge whereby all the horses square measure circled red. ClickAnimal may be a recognition-based CaRP theme engineered on top of Captcha menagerie [32], with associate alphabet of comparable animals such as dog, horse, pig, etc. Its positive identification may be a sequence of animal names like ρ = "Turkey, Cat, Horse, Dog,…." For each animal, one or additional 3D models square measure engineered. The Captcha generation method is applied to come up with ClickAnimal images: 3D models square measure wont to generate 2nd animals by applying completely different views, textures, colors, lightning effects, and optionally distortions. The ensuing 2nd animals square measure then arranged on a littered background like parcel. Some animals could also be occluded by different animals within the image, but their core elements aren't occluded so as for humans to spot each of them. Fig. four shows a click-animal image with associate alphabet of ten animals. Note that completely different views applied in mapping 3D models to 2nd animals, along with occlusion in the following step, turn out many alternative shapes for a similar animal's instantiations within the generated pictures. Combined with the extra anti-recognition mechanisms applied within the mapping step, these build it laborious for computers to acknowledge animals within the generated image, however humans will simply establish different instantiations of animals

C. AnimalGrid

The number of comparable animals is way but the quantity of available characters. ClickAnimal incorporates a smaller alphabet, and so a smaller positive identification house, than ClickText. CaRP should have a sufficiently-large effective positive identification house to resist human shot attacks. AnimalGrid's positive identification house can be inflated by combining it with a grid-based graphical password, with the grid reckoning on the scale of the chosen animal. DAS [3] may be a candidate however needs drawing on the grid. To be in line with ClickAnimal, we modify from drawing to clicking: Click-A-Secret (CAS) whereby a user clicks the grid cells in her positive identification. AnimalGrid may be a combination of ClickAnimal and CAS. the quantity of grid-cells in a very grid should be abundant larger than the alphabet size. Unlike DAS, grids in our CAS area unit object-dependent, as we are going to see next. It has the advantage that an accurate animal ought to 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 properly labelled grid-cell of the wrong grid would seemingly manufacture a wrong grid-cell at the authentication server facet once the right grid is employed. To enter a positive identification, a ClickAnimal image is displayed 1st. After AN animal is chosen, a picture of n × n grid seems, with the grid-cell size equalling the bounding parallelogram of the selected animal. every grid-cell is labelled to assist users establish. Fig. four shows a half dozen × half dozen grid once the red turkey within the left image of fig4 was chosen. A user will choose zero to multiple grid-cells matching her positive identification. so a positive identification may be a sequence of animals interleaving with grid-cells, e.g., ρ = "Dog, Grid2, Grid1; Cat, Horse, Grid3", wherever Grid1 means the grid-cell indexed as one, And grid-cells once an animal means that the grid is decided by the bounding parallelogram of the animal. A positive identification should begin with AN animal. When a ClickAnimal image seems, the user clicks the animal on the image that matches the primary animal in her password. The coordinates of the clicked purpose area unit recorded. The bounding parallelogram of the clicked animal is then found interactively as follows: a bounding parallelogram is calculated and displayed, e.g., the white parallelogram shown in Fig. 4. The user checks the displayed parallelogram and corrects inaccurate edges by dragging if required. This method is perennial till the user is satisfied with the accuracy of the bounding parallelogram. In most cases, the calculated bounding parallelogram is correct enough without needing manual correction. Once the bounding parallelogram of the chosen

animal is identified, a picture of n×n grid with the known bounding rectangle as its grid-cell size is generated and displayed. If the grid image is simply too massive or too tiny for a user to look at, the grid image is scaled to a fitting size. The user then clicks a sequence of zero to multiple grid-cells that match the grid cells following the primary animals in positive identification, then gets back to the ClickAnimal image. For the instance positive identification ρ given antecedently, she clicks some extent within grid-cell2, and then some extent within grid-cell1 to pick out the 2 grid-cells. The coordinates of user-clicked points on the grid image (the original one before scaling if the grid image is scaled) area unit recorded. The on top of method is perennial till the user has finished coming into her positive identification. The ensuing sequence of coordinates of user-clicked points, e.g., "AP150,50, GP30,66, GP89,160, AP135,97,…" wherever "APx,y" denotes the purpose with coordinates x,y on a ClickAnimal image, and "GPx,y" denotes the purpose with coordinates x,y on a grid image, is shipped to the authentication server. Using the bottom truth, the server recovers the primary animal from the received sequence, regenerates the grid image from the animal's bounding parallelogram, and recovers the clicked grid-cells. This method is perennial to recover the positive identification the user clicked. Its hash is then calculated and compared with the hold on hash.

V. RECOGNITION-RECALL CARP

In recognition-recall CaRP, a countersign could be a sequence of some invariant points of objects. associate invariant purpose of associate object (e.g. letter "A") could be a purpose that contains a mounted relative position in several incarnations (e.g., fonts) of the thing, and so will be unambiguously known by humans regardless of how the thing seems in CaRP pictures. The general public have a click variation of three pixels or less [18]. TextPoint, a recognition recall CaRP theme with associate alphabet of characters, is given next, followed by a variation for challenge response authentication

A. TextPoints

Characters contain invariant points. Fig. five shows some invariant points of letter "A", that offers a powerful cue to memorize and find its invariant points. a degree is claimed to be an indoor purpose of associate object if its distance to the closest boundary of the item exceeds a threshold. A set of internal invariant points of characters designated is chosen} to make a set of clickable points for TextPoints. The internality ensures that a clickable purpose is unlikely occluded by a neighboring character which its tolerance region unlikely overlaps with any tolerance region of a neighboring character's clickable points on the image generated by the underlying Captcha engine. In crucial clickable points, the gap between any combine of clickable points during a character should exceed a threshold so they're perceptually distinguishable and their tolerance regions don't overlap on CaRP pictures. In addition, variation ought to even be taken into thought. For example, if the middle of a stroke phase in one character is selected, we should always avoid choosing the middle of an analogous stroke phase in another character. Instead, we should always choose a different purpose from the stroke phase, e.g., a point at one-third length of the stroke phase to associate finish. This variation in choosing clickable purposes ensures that a clickable point is context-dependent: a equally structured purpose might or might not be a clickable purpose, betting on the character that the purpose lies in. Character recognition is needed in locating clickable points on a TextPoints image though the clickable points are illustrious for every character. this is often a task on the far side a bot's capability. Clickable points in the text points are salient points of their characters and so facilitate keep in mind a password, however can't be exploited by bots since they're each dynamic (as compared to static points in ancient graphical password schemes) and contextual:

- Dynamic: locations of clickable points and their contexts (i.e., characters) vary from one image to a different. The clickable points in one image area unit computationally freelance of the clickable points in another image.
- Contextual: whether or not a equally structured purpose may be clickable purpose or not depends on its context. It is only if at intervals the correct context, i.e., at the correct location of a right character.

These 2 options need recognizing the right contexts i.e., characters, first. By the terribly nature of Captcha, recognizing characters during a Captcha image may be a task on the far side computer's capability. Therefore, these salient points of characters cannot be exploited to mount lexicon attacks on TextPoints.

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

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 such as PassPoints [5]. 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 inde-pendent of the clickable points in another image
- **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.

These two features require recognizing the correct contexts, 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.

B. TextPoints4CR

For the CaRP schemes conferred up to currently, the coordinates of user-clicked points area unit sent on to the authentication server throughout authentication. For a lot of complicated protocols, say a challenge-response authentication protocol, a response is shipped to the authentication server instead. TextPoints may be changed to fit challenge-response authentication. This variation is termed TextPoints for Challenge-Response or TextPoints4CR. Unlike TextPoints whereby the authentication server stores a salt and a watchword hash worth for every account, the server in TextPoints4CR stores the watchword for every account. Another distinction is that every character seems just one occasion in a TextPoints4CR image however might seem multiple times in a TextPoints image. this is often as a result of each server and shopper in TextPoints4CR ought to generate a similar sequence of discredited grid-cells severally. that needs a novel way to generate the sequence from the shared secret, i.e., password. perennial characters would cause many potential sequences for a similar watchword. This distinctive sequence is used as if the shared secret during a typical challenge response authentication protocol. In TextPoints4CR, a picture is divided into a hard and fast grid with the discretization grid-cell of size μ on each directions. The negligible distance between any combine of clickable points should be larger than μ by a margin extraordinary a threshold to prevent 2 clickable points from falling into one grid-cell in a picture. Suppose that a secured tolerance of click errors on each coordinate axis and coordinate axis is τ , we require that $\mu \geq 4\tau$. Image Generation. to come up with a TextPoints4CR image, the same procedure to come up with a TextPoints image is applied. Then the subsequent procedure is applied to form each clickable point a minimum of τ distance from the perimeters of the grid-cell it lies in. All the clickable points denoted as set, area unit settled on the image.

VI. SECURITY ANALYSIS

A. Security of Underlying Captcha

Computational trait in recognizing objects in CaRP images is key to CaRP. Existing analyses on Captcha security were largely case by case or used AN approximate process. No suppositious security model has been established yet. Object segmentation is taken into account as a computationally expensive, combinatorial-hard downside [30], that fashionable text Captcha schemes deem. in line with [30], the quality of object segmentation, C, is exponentially dependent of the amount M of objects contained in a very challenge, and polynomially dependent of the dimensions N of the Captcha alphabet: $C = \alpha M P(N)$, wherever $\alpha >$ one may be a parameter, and P() may be a polynomial perform. A Captcha challenge usually contains 6 to ten characters, whereas a CaRP image usually contains 30 or a lot of characters. The quality to interrupt a Click-Text image is regarding $\alpha 30P(N)/(\alpha 10P(N)) = \alpha 20$ times the complexity to interrupt a Captcha challenge generated by its underlying Captcha theme. thus ClickText is way harder to interrupt than its underlying Captcha theme. moreover, characters in a very CaRP theme area unit organized two dimensionally further increasing segmentation issue due to one a lot of dimension to phase. As a result, we can reduce distortions in ClickText pictures for improved usability yet maintain constant security level because the underlying text Captcha.

B. Automatic Online Guessing Attacks

In automatic on-line guess attacks, the trial and error process is dead mechanically whereas dictionaries are constructed manually. If we tend to ignore negligible possibilities, CaRP with underlying CPA-Secure captcha has the subsequent properties:

1. Internal object-points on one CaRP image area unit computationally-independent of internal object-points on another CaRP image. notably, clickable points on one image area unit computationally-independent of clickable points on another image.
2. Eq. (3) holds, i.e., trials in guess attacks area unit reciprocally independent. The first property is verified by contradiction. Assume that the property doesn't hold, i.e., there exists an enclosed object-point α on one image A that's non-negligibly dependent of an enclosed object-point β on another image B. AN someone will exploit this

dependency to launch the following chosen-pixel attack. within the learning section, image A is employed to be told the item that contains purpose α. In the testing section, purpose β on image B is employed to question the oracle. Since purpose α is non-negligibly dependent of purpose β, this CPA-experiment would end in successful chance non-negligibly higher than a random guess, that contradicts the CPA-secure assumption. we tend to conclude that the primary property holds. The second property may be a consequence of the primary property since user-clicked internal object-points in one trial area unit computationally-independent of user-clicked internal object-points in another trial as a result of the primary property. We have ignored background and boundary object-points since clicking any of them would cause authentication failure. Eq. (3) indicates that automatic on-line guess attacks can notice a word solely probabilistically regardless of however many trials area unit dead.

C. Human Guessing Attacks

In human guess attacks, humans area unit accustomed enter passwords within the trial and error method. Humans area unit abundant slower than computers in mounting guess attacks. For 8-character passwords, the theoretical word house is 338 ≈ 240 for ClickText with AN alphabet of thirty three characters, 108 ≈ 226 for ClickAnimal with AN alphabet of ten animals, and ten × 467 ≈ 242 for AnimalGrid with the setting as ClickAnimal and 6×6 grids. If we tend to assume that one thousand folks are utilized to figure eight hours per day with none stop in a very human guess attack, which every person takes thirty seconds to finish one trial. it might take them the typical} zero.5 · 338 ·30/ (3600 · eight · one thousand · 365) ≈ 2007 years to interrupt a ClickText password, 0.5 · 108 · 30/(3600 · eight · 1000) ≈ five2 days to interrupt a ClickAnimal word, or 0.5 · ten · 467 · 30/(3600 · eight ·1000 · 365) ≈ 6219 years to interrupt AN AnimalGrid word. Human guess attacks on TextPoints need a way longer time than those on ClickText since TextPoints features a abundant larger word house. Just like any word theme, a longitudinal analysis is needed to ascertain the effective word house for every CaRP mental representation. this needs a separate study kind of like what Bonnie [42] did for text passwords. A recent study on text passwords [42] indicates that users tend to decide on passwords of 6–8 characters and have a strong dislike of exploitation non-alphanumeric characters, and that an acceptable benchmark of effective word house is that the expected variety of best guesses per account required to break five hundredth of accounts, that is like twenty one.6 bits for Yahoo! users. If we tend to assume that ClickText has roughly the same effective word house as text passwords, it needs on average one thousand folks to figure one.65 days or one person to work 4.54 years to seek out a ClickText word.

D. Relay Attacks

Relay attacks is also dead in many ways that. Captcha challenges is relayed to a high-volume web site hacked or controlled by adversaries to possess human surfers solve the challenges so as to continue surfboarding the web site, or relayed to sweatshops wherever humans area unit employed to unravel Captcha challenges for little payments. Is CaRP susceptible to relay attacks? we tend to create constant assumption as Van Oorschot and Stubblebine [15] in discussing CbPA-protocol's lustiness to relay attacks: someone won't deliberately participate in relay attacks unless obtained the task. The task to perform and also the image employed in CaRP area unit terribly totally different from those accustomed solve a Captcha challenge. This noticeable distinction makes it exhausting for someone to erroneously facilitate check a word guess by attempting to unravel a Captcha challenge.

E. Shoulder-Surfing Attacks

Shoulder-surfing attacks area unit a threat once graphical passwords area unit entered in a very public place like bank ATM machines. CaRP isn't sturdy to shoulder-surfing attacks by itself. However, combined with the subsequent dual-view technology, CaRP will thwart shoulder-surfing attacks. By exploiting the technical limitation that commonly-used LCDs show varied brightness and color reckoning on the viewing angle, the dual-view technology will use code alone to show 2 pictures on a alphanumeric display screen at the same time, one public image see able at the most view-angles, and also the different private image see able solely at a selected view-angle [38]. When a CaRP image is displayed because the "private" image by the dual-view system, a shoulder-surfing offender will capture user clicked points on the screen, however cannot capture the "private" CaRP image that solely the user will see. However, the obtained user-clicked points area unit useless for an additional login try, where a new, computationally-independent image are used and thus the captured points won't represent the proper word on the new image any longer. To the contrary, common implementations of graphical password schemes like PassPoints use a static input image in the same location of the screen for every login try. Although this image is hidden because the personal image by the dual-view technology from being captured by a shoulder-surfer, the user-clicked points captured in a very winning login are still the valid word for next login try. That is, capturing the points alone is decent for an efficient attack in this case. In general, the upper the correlation of user-clicked points between totally different login makes an attempt is, the less effective protection the dual-view technology would supply to thwart shoulder-surfing attacks.

F. Others

CaRP isn't bulletproof to any or all potential attacks. CaRP is vulnerable if a consumer is compromised specified each the image and user-clicked points is captured. Like several different graphical passwords like CCP and PCCP, CaRP schemes using the essential CaRP authentication area unit susceptible to phishing since user-clicked points area unit sent to the authentication server. However, CaRP schemes like TextPoints4CR used with challenge-response authentication area unit sturdy to phishing to a certain level: a phishing someone should mount offline guessing attacks to seek out the word exploitation the verifiable data obtained through winning phishing attack

VII. CONCLUSION

We have projected CaRP, a replacement security primitive relying on unsolved exhausting AI issues. CaRP is each a Captcha and a graphical word theme. The notion of CaRP introduces a new family of graphical passwords, that adopts a new approach to counter on-line guess attacks: a replacement CaRP image, that is additionally a Captcha challenge, is used for every login arrange to create trials of an internet guess attack computationally freelance of every different. A password of CaRP is found solely probabilistically by automatic online guess attacks together with brute-force attacks, a desired security property that different graphical word schemes lack. Hotspots in CaRP pictures will now not be exploited to mount automatic on-line guess attacks, AN inherent vulnerability in many graphical word systems. CaRP forces adversaries to resort to considerably less economical and far a lot of pricey human-based attacks. additionally to protectively from online guess attacks, CaRP is additionally proof against Captcha relay attacks, and, if combined with dual-view technologies, shoulder-surfing attacks. CaRP may also facilitate scale back spam emails sent from an online email service. Our usability study of 2 CaRP schemes we've implemented is encouraging. for instance, a lot of participants considered AnimalGrid and ClickText easier to use than PassPoints and a mixture of text word and Captcha. Both AnimalGrid and ClickText had higher word memorability than the standard text passwords. On the opposite hand, the usability of CaRP is additional improved by exploitation pictures of different levels of issue supported the login history of the user and also the machine accustomed log in. The best exchange between security and value remains AN open question for CaRP, and additional studies area unit required to refine CaRP for actual deployments. Like Captcha, CaRP utilizes unsolved AI issues. However, a word is way a lot of valuable to attackers than a free email account that Captcha is usually accustomed shield. Therefore there are a unit a lot of incentives for attackers to hack CaRP than Captcha. That is, a lot of efforts are drawn to the following win-win game by CaRP than standard Captcha: If attackers succeed, they contribute to rising AI by providing solutions to open issues like segmenting 2D texts. Otherwise, our system stays secure, tributary to sensible security. As a framework, CaRP doesn't trust on any specific Captcha theme. once one Captcha theme is broken, a replacement and safer one might seem and be converted to a CaRP theme.

Overall, our work is one leap forward within the paradigm of using exhausting AI issues for security. Of affordable security and usability and sensible applications, CaRP has sensible potential for refinements, that imply helpful future work. More significantly, we tend to expect CaRP to inspire new inventions of such AI based mostly security primitives.

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] (2012, Feb.). *The Science Behind Passfaces* [Online]. Available: http://www.realuser.com/published/ScienceBehindPassfaces.pdf

[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] S. 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. 63, pp. 102–127, Jul. 2005.

[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 TippingPoint DVLabs, Vienna, Austria. (2010). *Top Cyber Security Risks Report, SANS Institute and Qualys Research Labs* [Online]. Available: http://dvlabs.tippingpoint.com/toprisks2010

[14] B. Pinkas and T. Sander, "Securing passwords against dictionary attacks," in *Proc. ACM CCS*, 2002, pp. 161–170.

[15] 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. 9, no. 3, pp. 235–258, 2006.

[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.

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

[19] 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.

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

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

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

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

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

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