SET BASED CAPTCHA- A NEW TECHNIQUE TO TACKLE WEB ROBOTS

Asha Pahuja[1], Dayanand[2], Vishal Srivastav[3]

[1]M.Tech (CSE) Student, [3]Professor, Dept. of CSE & IT, Arya College of Engineering and &IT. Jaipur, (India)

[2]Assistant Professor Dept. of CSE, HMR Institute of Tech. & Mgt. New Delhi, (India)

ABSTRACT

"CAPTCHA" stands for Completely Automated Public Turing Test [1] to Tell Computers and Humans Apart [2]. As use of internet has been increased, security of web application has become a vital issue and many web applications facing a threat of web bots also known as internet Robot Web-bots/ Internet Robot is an automated script which executes over the web forms and occupy web spaces and thus increases network traffic. The issue with the most used captcha i.e. text based captcha is that most of them have proven to be either not robust enough (they have been broken) or they are too complicated or annoying to read even for humans. Image based Captcha in various type has been used but proven to be broken many times .Set Based Captcha is a type of captcha in which user has to exactly match with words displayed in two sets. This paper discusses various existing captcha and Set Based Captcha. This paper also evaluates different captcha based on evaluation parameters.

Keywords: CAPTCHA, Text Based captcha, Image based captcha, Set Based Captcha.

I. INTRODUCTION

You're trying to sign up for a free email service offered by Gmail or Yahoo. Before you can submit your application, you first have to pass a test. It's not a hard test -- in fact, that's the point [1]. For you, the test should be simple and straightforward. But for a computer, the test should be almost impossible to solve.

This sort of test is a CAPTCHA. They're also known as a type of Human Interaction Proof (HIP). You've probably seen CAPTCHA tests on lots of Web sites. The most common form of CAPTCHA is an image of several distorted letters. It's your job to type the correct series of letters into a form. If your letters match the ones in the distorted image, you pass the test.

CAPTCHAs are short for Completely Automated Public Turing test to tell Computers and Humans Apart. The term "CAPTCHA" was coined in 2000 by Luis Von Ahn, Manuel Blum, Nicholas J. Hopper (all of Carnegie Mellon University, and John Langford (then of IBM). They are challenge-response tests to ensure that the users are indeed human. The purpose of a CAPTCHA is to block form submissions from spam bots – automated scripts that harvest email addresses from publicly available web forms. A common kind of CAPTCHA used on most websites requires the users to enter the string of characters that appear in a distorted form on the screen.

CAPTCHAs are used because of the fact that it is difficult for the computers to extract the text from such a distorted image, whereas it is relatively easy for a human to understand the text hidden behind the distortions. Therefore, the correct response to a CAPTCHA challenge is assumed to come from a human and the user is permitted into the website.

Why would anyone need to create a test that can tell humans and computers apart? It's because of people trying to game the system -- they want to exploit weaknesses in the computers running the site. While these individuals probably make up a minority of all the people on the Internet, their actions can affect millions of users and Web sites.

For example, a free e-mail service might find itself bombarded by account requests from an automated program. That automated program could be part of a larger attempt to send out spam mail to millions of people. The CAPTCHA test helps identify which users are real human beings and which ones are computer programs.

Spammers are constantly trying to build algorithms that read the distorted text correctly. So strong CAPTCHAs have to be designed and built so that the efforts of the spammers are thwarted.

II. MOTIVATION

The proliferation of the publicly available services on the Web is a boon for the community at large. But unfortunately it has invited new and novel abuses. Programs (bots and spiders) are being created to steal services and to conduct fraudulent transactions. Some examples:

Free online accounts are being registered automatically many times and are being used to distribute stolen or copyrighted material.

Recommendation systems are vulnerable to artificial inflation or deflation of rankings. For example, EBay, a famous auction website allows users to rate a product. Abusers can easily create bots that could increase or decrease the rating of a specific product, possibly changing people’s perception towards the product.

Spammers register themselves with free email accounts such as those provided by Gmail or Hotmail and use their bots to send unsolicited mails to other users of that email service.

Online polls are attacked by bots and are susceptible to ballot stuffing. This gives unfair mileage to those that benefit from it.

In light of the above listed abuses and much more, a need was felt for a facility that checks users and allows access to services to only human users. It was in this direction that such a tool like CAPTCHA was created.

III. BACKGROUND

The need for CAPTCHAs rose to keep out the website/search engine abuse by bots. In 1997, AltaVista sought ways to block and discourage the automatic submissions of URLs into their search engines. Andrei Broder, Chief Scientist of AltaVista, and his colleagues developed a filter. Their method was to generate a printed text randomly that only humans could read and not machine readers. Their approach was so effective that in a year, “spam-add-ons’” were reduced by 95% and a patent was issued in 2001.

In 2000, Yahoo’s popular Messenger chat service was hit by bots which pointed advertising links to annoying human users of chat rooms. Yahoo, along with Carnegie Mellon University, developed a CAPTCHA called EZ-

GIMPY, which chose a dictionary word randomly and distorted it with a wide variety of image occlusions and asked the user to input the distorted word.

In November 1999, slashdot.com released a poll to vote for the best CS College in the US. Students from the Carnegie Mellon University and the Massachusetts Institute of Technology created bots that repeatedly voted for their respective colleges. This incident created the urge to use CAPTCHAs for such online polls to ensure that only human users are able to take part in the polls.

IV. CAPTCHAS AND THE TURING TEST

CAPTCHA technology has its foundation in an experiment called the **Turing Test**. Alan Turing, sometimes called the father of modern computing, proposed the test as a way to examine whether or not machines can think -- or appear to think -- like humans. The classic test is a game of imitation. In this game, an interrogator asks two participants a series of questions. One of the participants is a machine and the other is a human. The interrogator can't see or hear the participants and has no way of knowing which is which. If the interrogator is unable to figure out which participant is a machine based on the responses, the machine passes the Turing Test.

Of course, with a CAPTCHA, the goal is to create a test that humans can pass easily but machines can't.

It's also important that the CAPTCHA application is able to present different CAPTCHAs to different users. If a visual CAPTCHA presented a static image that was the same for every user, it wouldn't take long before a spammer spotted the form, deciphered the letters, and programmed an application to type in the correct answer automatically.

Most, but not all, CAPTCHAs rely on a visual test. Computers lack the sophistication that human beings have when it comes to processing visual data. We can look at an image and pick out patterns more easily than a computer. The human mind sometimes perceives patterns even when none exist, a quirk we call pareidolia. Ever see a shape in the clouds or a face on the moon? That's your brain trying to associate random information into patterns and shapes.

But not all CAPTCHAs rely on visual patterns. In fact, it's important to have an alternative to a visual CAPTCHA. Otherwise, the Web site administrator runs the risk of disenfranchising any Web user who has a visual impairment. One alternative to a visual test is an audible one. An audio CAPTCHA usually presents the user with a series of spoken letters or numbers. It's not unusual for the program to distort the speaker's voice, and it's also common for the program to include background noise in the recording. This helps thwart voice recognition programs.

Another option is to create a CAPTCHA that asks the reader to interpret a short passage of text. A contextual CAPTCHA quizzes the reader and tests comprehension skills. While computer programs can pick out key words in text passages, they aren't very good at understanding what those words actually mean.

V. FORMS OF ATTACKS

Whether a captcha is based on pictures, text, sound, or puzzle–solving, certain similarities can be seen in terms of how captchas are attacked by malicious users. Typical attack models seen to date include:

5.1 Bypass Attacks

Any attack that circumvents the need to solve the captcha at all. For example, network replay attacks, or cases where the captcha solution is exposed accidentally, perhaps through HTML or CGI form parameter values. Generally, any system that sends the decoded form of the captcha to a client program as part of the data stream is vulnerable to such an attack. Such attacks are not always a weakness of the captcha itself; they may instead be a weakness of the service using the captcha.

5.2 Challenge Replay Attacks

If the captcha system can produce only a limited number of unique challenges, then the automated agent may record all or most of the possible challenges. A human associate provides a library of correct answers for the challenges. The automated agent can then replay the correct answer whenever it is faced with a particular challenge for which it knows the correct solution. Some image–based captchas are vulnerable to this weakness, particularly those based upon a finite library of photographs (e.g., the ‘KittenAuth’ captcha (Reimer, 2006) used a challenge library of 42 images).

Signal processing attacks

The noise and perturbations that are commonly used to obfuscate captcha images or sounds are intended to be one–way; a computer should be able to add them, but not reverse them easily. In principle, only a human’s flexible image and sound recognition capabilities should be capable of conveniently reversing the transformations and recovering the original message. In practice, captcha researchers and malicious attackers have both proven highly capable of reversing captcha transformations approximately via mathematical heuristics and machine learning approaches (Chellapilla and Simard, 2005; Hocevar, 2004; Huang, et al., 2008; Mori and Malik, 2003; Yan and El Ahmad, 2007, 2008a). For text–based captchas, this is achieved by removing image noise and clutter items, and isolating individual characters within the captcha in order to allow optical character recognition (OCR) technologies a maximised opportunity for success. Attacking heuristics often have a parameterised design, so that their behaviour may be adjusted to attack several different but related forms of captcha.

5.3 Mechanical Turk Attacks

Here, the problem of solving the captcha is automatically ‘outsourced’ to a paid human agent. They immediately solve the challenge and quickly return the answer to the automated agent in real time. The automated agent then presents the human–provided answer, and is able to programatically exploit the online resource (Barr and Cabrera, 2006; Bursztein, et al., 2010; Economist, 2008; Websense Security Labs, 2008b). A human ‘Turk’ agent working full time to support such attacks can solve thousands of captchas per hour, depending on the type of captcha. There is little that can be done to defend against such attacks, other than to perhaps raise the inconvenience of the captcha for all users in order to reduce the economic viability of this attack. Generally, an increase in inconvenience can be achieved either by increasing the difficulty of the captcha for humans, or by requiring users to regularly re–authenticate themselves as human.

5.4 Trivial Guessing Attacks

If there is an unlimited range of challenges, but a very limited range of possible answers (e.g., 'which of these 10 choices is correct?'), a high success rate may be achieved by an attacking program by merely guessing randomly from the available answers. Particularly, any graphical captcha that requires the user to select a correct position within an image — but which has a wide error tolerance for user inaccuracy — may be vulnerable to a trivial guessing attack.

Brute force attacks

If there is a somewhat limited range of possible answers — e.g., a numerical 4–digit captcha would have 10,000 possible answers — then it is possible for a distributed group of automated agents to attack the captcha by exhaustively trying answers at random or according to a selected sequence. This differs from the 'trivial guessing attack', in that it relies upon having access to a large number of attacking agents — i.e., a 'botnet' (Websense Security Labs, 2008b; 2009) — rather than relying upon having access to a poorly designed captcha.

5.5 Hybrid Attacks

It is possible to combine these attacks. For example, if a signal processing attack can estimate five of six captcha characters with a high degree of confidence, a guess may be made on the remaining character, yielding a success rate of between 1.5% (mixed case alphanumerical characters) and 10% (numerical digits). For example, the 'Question–Based captcha' (Shirali–Shahreza and Shirali–Shahreza, 2007b) presents a mathematical problem, which can be broken by an attacker who uses OCR to recognise the numerical digits mentioned in the puzzle, combined with a random guess of one of the few possible ways in which the numbers may be combined arithmetically.

CAPTCHAs are classified based on what is distorted and presented as a challenge to the user. They are:

VI. TYPE OF CAPTCHAS

6.1 Text Captcha

These are simple to implement. The simplest yet novel approach is to present the user with some questions which only a human user can solve. Examples of such questions are:

What are twenty minus three?

What is the third letter in UNIVERSITY?

Which of Yellow, Thursday and Richard is a color?

If yesterday was a Sunday, what is today?

Such questions are very easy for a human user to solve, but it's very difficult to program a computer to solve them. These are also friendly to people with visual disability – such as those with colour blindness.

Other text CAPTCHAs involves text distortions and the user is asked to identify the text hidden. The various implementations are:

6.1.1 Gimpy

Gimpy is a very reliable text CAPTCHA built by CMU in collaboration with Yahoo for their Messenger service. Gimpy is based on the human ability to read extremely distorted text and the inability of computer programs to do the same. Gimpy works by choosing ten words randomly from a dictionary, and displaying them in a distorted and overlapped manner. Gimpy then asks the users to enter a subset of the words in the image. The human user is capable of identifying the words correctly, whereas a computer program cannot.

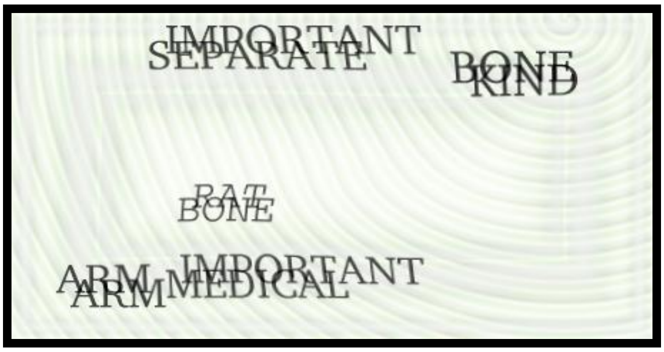


Fig 2.1 Gimpy CAPTCHA

6.1.2 Ez – Gimpy

This is a simplified version of the Gimpy CAPTCHA, adopted by Yahoo in their signup page. Ez – Gimpy randomly picks a single word from a dictionary and applies distortion to the text. The user is then asked to identify the text correctly.

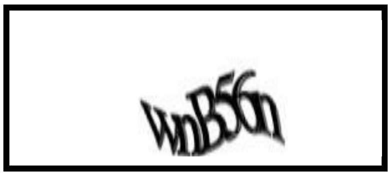


Fig 2.2 Yahoo's Ez – Gimpy CAPTCHA

6.1.3 Baffle Text

This was developed by Henry Baird at University of California at Berkeley. This is a variation of the Gimpy. This doesn't contain dictionary words, but it picks up random alphabets to create a nonsense but pronounceable text. Distortions are then added to this text and the user is challenged to guess the right word.

This technique overcomes the drawback of Gimpy CAPTCHA because, Gimpy uses dictionary words and hence, clever bots could be designed to check the dictionary for the matching word by brute-force.

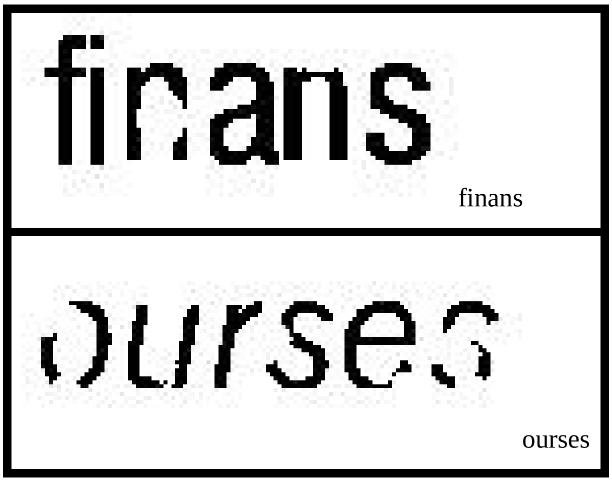
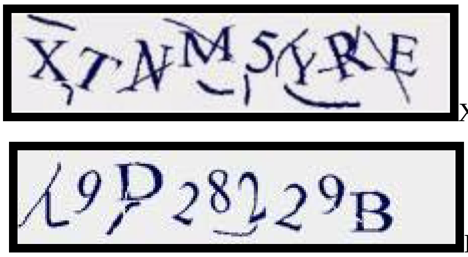


Fig 2.3 Baffle Text examples

6.1.4 MSN Captcha

Microsoft uses a different CAPTCHA for services provided under MSN umbrella. These are popularly called MSN Passport CAPTCHAs. They use eight characters (upper case) and digits. Foreground is dark blue, and background is grey. Warping is used to distort the characters, to produce a ripple effect, which makes computer recognition very difficult.



XTNM5YRE

L9D28229B

Fig 2.4 MSN Passport CAPTCHA

6.2 Graphic CAPTCHAs

Graphic CAPTCHAs are challenges that involve pictures or objects that have some sort of similarity that the users have to guess. They are visual puzzles, similar to Mensa tests. Computer generates the puzzles and grades the answers, but is itself unable to solve it.

6.2.1 Bongo

Bongo. Another example of a CAPTCHA is the program we call BONGO [2]. BONGO is named after M.M. Bongard, who published a book of pattern recognition problems in the 1970s [3]. BONGO asks the user to solve a visual pattern recognition problem. It displays two series of blocks, the left and the right. The blocks in the left series differ from those in the right, and the user must find the characteristic that sets them apart. A possible left and right series is shown in Figure 2.5

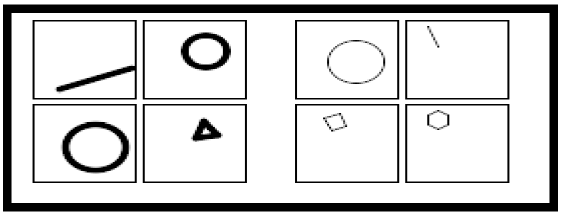


Fig 2.5 Bongo CAPTCHA

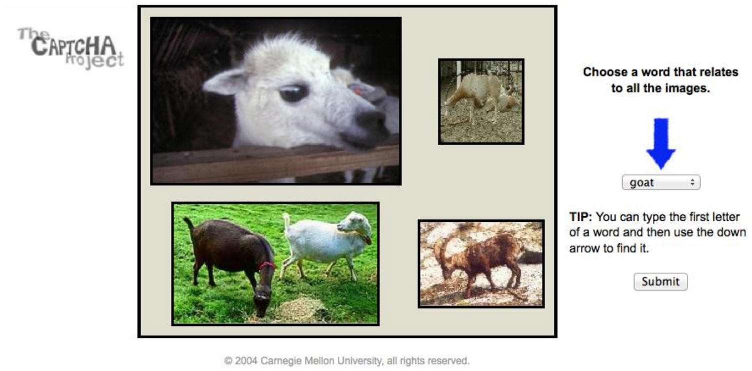
These two sets are different because everything on the left is drawn with thick lines and those on the right are in thin lines. After seeing the two blocks, the user is presented with a set of four single blocks and is asked to determine to which group the each block belongs to. The user passes the test if s/he determines correctly to which set the blocks belong to. We have to be careful to see that the user is not confused by a large number of choices.

6.2.2 Image Based Captcha

Image based CAPTCHA: In this scheme the user is required to identify some image recognition task.

6.2.3 ESP Pix

ESP Pix is first image CAPTCHA and it was developed at Carnegie Mellon University [13]. A snapshot of ESP Pix CAPTCHA is shown in Figure 2. In ESP Pix the user has given four images and in order to pass this test the user has to select word related to those four images from drop down list of 72 choices.



Choose a word that relates to all the images.

goat

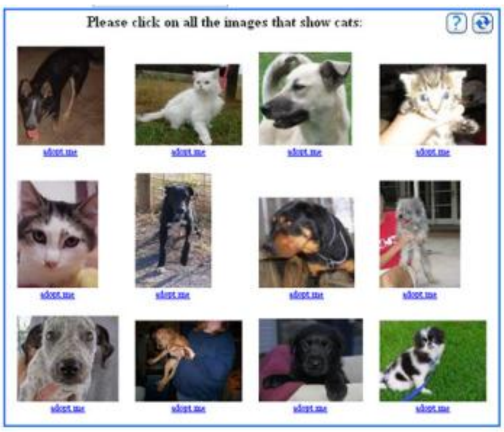
TIP: You can type the first letter of a word and then use the down arrow to find it.

Submit

© 2004 Carnegie Mellon University, all rights reserved.

6.2.4 Asirra

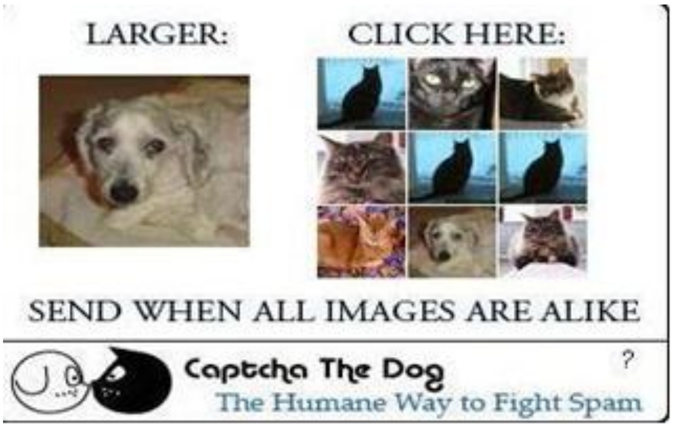
Another Image CAPTCHA is Asirra Stands for Animal Species Image Recognition for Restricting Access is a cat or dog labeling based CAPTCHA design [14]. In this test user has to select all the pics of cat. Asirra is randomly choosing images from petfinder.com. Snapshot of Asirra is shown in figure 3.



Please click on all the images that show cats:

6.2.5 CAPTCHA the Dog

One CAPTCHA is available as paid service is CAPTCHA the dog [14]. It Shows nine images in 3 by 3 grid and user is asked to chose all the images of cat one by one until images become dog‟s images. A snapshot of it is shown in Figure 4.The dog is randomly placed among nine cats and the process is repeated for three times.



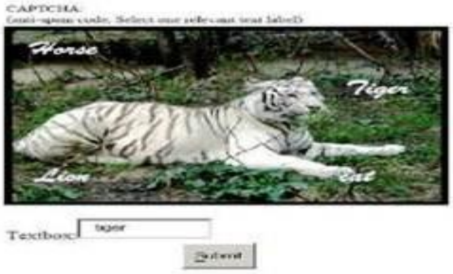
LARGER:

CLICK HERE:

SEND WHEN ALL IMAGES ARE ALIKE

Multi Model CAPTCHA

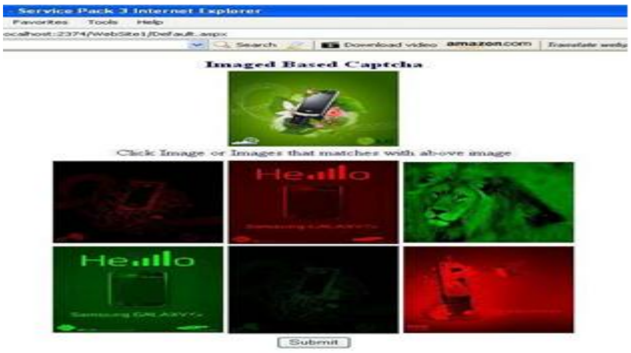
It combines text and image based System together. In this end user is shown an image and four text labels associated with the image. Text labels are embedded in the image and the user is asked to select a relevant text label [15]. A snapshot of Multi Model CAPTCHA is shown in Figure 5



CAPTCHA:
(anti-spam code, Select one relevant text label)

6.2.6 Dynamic Image Based CAPTCHA

Another improved image CAPTCHA is Dynamic Image Based CAPTCHA (DIBC).In this CAPTCHA system user is required to recognize the exact matching image or images to pass the Turing test. An image is selected randomly from image database and is placed in a grid of six images random number of times. User is supposed to submit all the correct version of the filtered image for clearing the Turing test in maximum of 5 attempts [16].It is shown in Figure 6

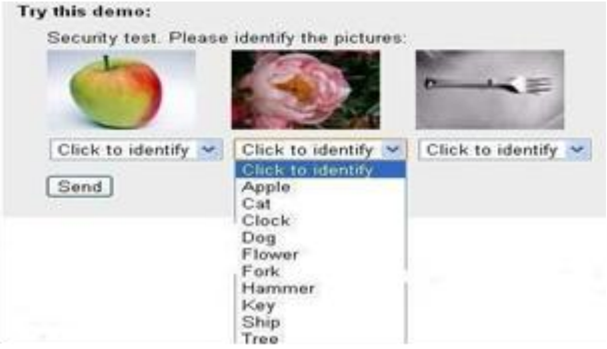


Imaged Based Captcha

Click Image or Images that matches with above image

6.2.7 IdentiPic

It is photo based CAPTCHA system where user has to identify picture [17].Three pictures are shown and corresponding to each pic there is drop down list having ten options. A snapshot of Identipic is shown in Figure 7.



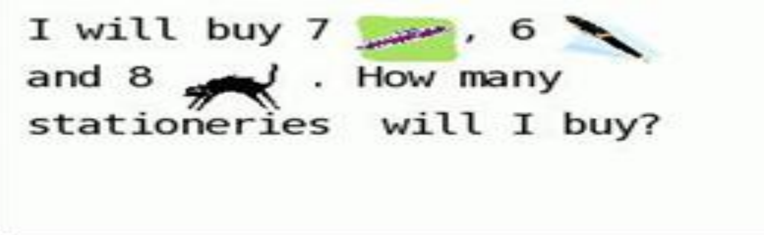
Try this demo:

Security test. Please identify the pictures:

6.2.8 Puzzle Based CAPTCHA

It is also referred as question based CAPTCHA [18]. In this test, a small mathematical problem is generated according to some predefined rules. The problem then rendered by the server to the user answer of which is already known to server. Solving of this problem requires an ability of understanding text of question, only a human user can answer this question.

Figure 8 illustrates the GUI of Question based CAPTCHA



I will buy 7 , 6 and 8 . How many stationeries will I buy?

6.3 Audio CAPTCHAs

The final example we offer is based on sound. The program picks a word or a sequence of numbers at random, renders the word or the numbers into a sound clip and distorts the sound clip; it then presents the distorted sound clip to the user and asks users to enter its contents. This CAPTCHA is based on the difference in ability between humans and computers in recognizing spoken language. Nancy Chan of the City University in Hong Kong was the first to implement a sound-based system of this type. The idea is that a human is able to efficiently disregard the distortion and interpret the characters being read out while software would struggle with the distortion being applied, and need to be effective at speech to text translation in order to be successful. This is a crude way to filter humans and it is not so popular because the user has to understand the language and the accent in which the sound clip is recorded.

6.4 re CAPTCHA and Book Digitization

To counter various drawbacks of the existing implementations, researchers at CMU developed a redesigned CAPTCHA aptly called the reCAPTCHA. About 200 million CAPTCHAs are solved by humans around the world every day. In each case, roughly ten seconds of human time are being spent. Individually, that's not a lot of time, but in aggregate these little puzzles consume more than 150,000 hours of work each day. What if we could make positive use of this human effort? reCAPTCHA does exactly that by channeling the effort spent solving CAPTCHAs online into "reading" books.

To archive human knowledge and to make information more accessible to the world, multiple projects are currently digitizing physical books that were written before the computer age. The book pages are being photographically scanned, and then transformed into text using "Optical Character Recognition" (OCR). The transformation into text is useful because scanning a book produces images, which are difficult to store on small devices, expensive to download, and cannot be searched. The problem is that OCR is not perfect.

reCAPTCHA improves the process of digitizing books by sending words that cannot be read by computers to the Web in the form of CAPTCHAs for humans to decipher. More specifically, each word that cannot be read correctly by OCR is placed on an image and used as a CAPTCHA. This is possible because most OCR programs alert you when a word cannot be read correctly.

But if a computer can't read such a CAPTCHA, how does the system know the correct answer to the puzzle? Here's how: Each new word that cannot be read correctly by OCR is given to a user in conjunction with another word for which the answer is already known. The user is then asked to read both words. If they solve the one for which the answer is known, the system assumes their answer is correct for the new one. The system then gives the new image to a number of other people to determine, with higher confidence, whether the original answer was correct

Currently, reCAPTCHA is employed in digitizing books as part of the Google Books Project.

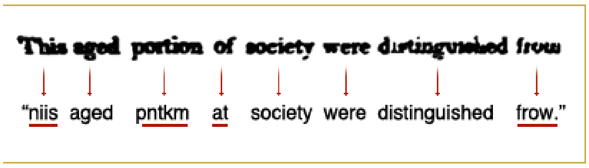
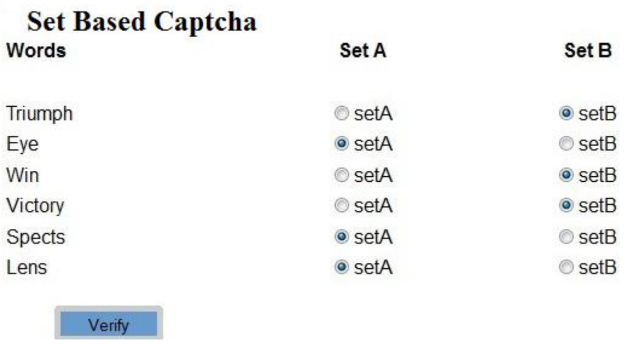

This aged portion of society were distinguished from
“niis aged pntkm at society were distinguished frow.”

Fig 2.6 First Line Shows Scanned Text, Second Line Shows Text Read by OCR

6.5 Proposed System

6.5.1 Set Based Captcha The user is presented with six words, is asked to divide the group into two subsets. Any categorization is possible, for example: do they sound same? Whether three words belongs to games while other three are computer gadgets. There will be three column, in which first column will show words, second column will show radio buttons of set A and third column will show radio buttons of set B. There will be six rows which will represent all six words. User just needs to click on radio buttons by dividing those six words in two groups as set A and set B.



Set Based Captcha

Words	Set A	Set B
Triumph	○ setA	● setB
Eye	● setA	○ setB
Win	○ setA	● setB
Victory	○ setA	● setB
Spects	● setA	○ setB
Lens	● setA	○ setB

6.5.2 Advantage Over Existing Captcha System

No readability issue with Set Based captcha. Words are easily understandable, no confusion in recognising them. We can add large number of words setssets in our database. No problem with people having colour vision problem. User just needs to divide the words in two words sets. It only requires text based interface. As it is new in comparison with existing captcha system so attacks are less vulnerable.

VII. APPLICATIONS

CAPTCHAs are used in various Web applications to identify human users and to restrict access to them.

Some of them are:

1. Online Polls
2. Protecting Web Registration:
3. Preventing comment spam
4. Search engine bots:
5. E-Ticketing: Email spam
6. Preventing Dictionary AttacksAs a tool to verify digitized books:
7. Improve Artificial Intelligence (AI) technology:

VIII. CONCLUSION

Creating a captcha that is so secure that no human cansolve it or so user friendly that it is a trivial task forcaptcha breaking software is very easy to accomplish.A successful captcha by its definition is able to tellhumans and computers a part. The goal is to addsecurity features whenever possible as long as they donot significantly or unnecessarily decrease theaccuracy of human solvers. Text based captcha arenow breakable. If we will increase distortion, blurringand other factors, then it will be hard for humanbeings also to read those texts while our goal is todifferentiate between humans and computers. Set Based captcha proposes a solution for thisproblem. It’s hard to break and user may not find anydifficulty in dividing those words in two words sets.Since user has to just click on radio buttons, so it isless time consuming also.

IX. FUTURE WORK

Usability issues with Set Based Captcha. The issues may be difficulty level of words in Word Sets, Instead of dividing those six words in twowords sets by just clicking on radio button and submit, we can ask user to write those six words in two words sets. A lot of work is needed forConsistency evaluation. How much entropy isactually present in each Captcha, as a way ofdetermining how vulnerable they are to guessing attacks?

REFRENCES

[1]. MoniNaor “Verification of a human in the loop oridentification via the Turing test”. Unpublished

[2]. Manuscript,1997.A preliminary draft available on

[3]. http://www.wisdom.weizmann.ac.il/~naor/PAPERS/human.pdf Pages. 2-3.

[4]. Luis Von Ahn , Manuel Blum , and Jo n Langford “Tellinghumans and computer apart automatically”. Incommunications of the ACM Vol. 47, No. 2, February’2004,Pages. 57-58

[5]. Graeme Baxter Bell “Strengthening Captcha based websecurity” ,First Monday, Volume 17, Number 2 - 6 February2012 Pages 2-3

[6]. http://firstmonday.org/ojs/index.php/fm/article/viewArticle/3630/3145

[7]. http://www.informationweek.com/news/internet/webdev/showArticle.jhtml?articleID=205900620

[8]. http://www.theregister.co.uk/2008/02/25/gmailcaptchacrack/

[9]. http://blogs.zdnet.com/security/?p=1232&tag=nl.e550

[10]. Rizwan Ur Rahman “Survey on Captcha systems” InJournal of Global Research in Computer Science, Volume 3,

[11]. No. 6, June 2012, Pages. 55-57 identiPic CAPTCHAavailable at http://identipic.com

[12]. Moin Mahmud Tanvee, Mir TafseerNayeem, Md.MahmudulHasanRafee ”Move & Select: 2-LayerCAPTCHA based on Cognitive Psychology for securingweb services”,Taken from ijens.org, available athttp://www.ijens.org/ Vol_11_I _05/117005-8383-IJVIPNSIJENS.pdf.

[13]. ElieBursztein, Matthieu Martin, John C. Mitchell ”TextbasedCAPTCHA Strengths and Weaknesses” in ACMComputer and Communication security 2011 (CSS’2011)Pages 11,12

[14]. E. Bursztein and S. Bethard, 2009. “Decaptcha: Breaking75% of eBay audio CAPTCHAs,” Proceedings of WOOT’09: Third USENIC Workshop on Offensive Technologies (10August Montreal, Canada), athttp://www.usenix.org/event/woot09/tech/full_papers/bursztein.pdf accessed 10 February 2012. Page-3

[15]. Jennifer Tam, Jiri Simsa, Sean Hyde, Luis Von Ahn“BreakingAudio CAPTCHAs” Carnegie Mellon University page -

[16]. 5,www.captcha.net/Breaking_Audio_CAPTCHAs.pdf

[17]. AndrewKirillov. aforge framework.http://www.aforgenet.com/framework/

[18]. C.Cortesand V. Vapnik.” Support vector networks. Machinelearning”, 20(3):273–297, 1995

[19]. J.Yan and A.S. El Ahmad. “A Low-cost Attack on a MicrosoftCAPTCHA”. In Proceedings of the 15th ACM conferenceon Computer and communications security, pages 543–554.ACM,2008.

[20]. Wikipedia. Flood fill algorithm.http://en.wikipedia.org/wiki/Flood_fill

[21]. WaseemDaher “POSH: A Generalized Captcha withsecurity application” In AISec ,Proceedings of the 1st ACMworkshop on Workshop on AISec ,2008,Pages. 2-10

[22]. Luis Von Abn, Manual Blum, Nichlas Hoper and JohnLangford CAPTCHA: “Using Hard AI Problems For Security”. Computer Science Dept., Carnegie MellonUniversity, Pittsburgh PA 15213, USA

[23]. H. S. Baird and K. Popat.“Human interactive proofs and document image analysis”. Proc. of 5th IAPR Int.Workshop on Document Analysis Systems (DAS 2002), vol.2423 of LNCS, pp. 507–518, 2002

[24]. The Official CAPTCHA site located on the internet athttp://www.captcha.net/

[25]. Dictionary Attack, available at

[26]. http://en.wikipedia.org/wiki/Dictionary_attack.

[27]. S. Chakrabarti and M. Singhal.“Password-based authentication: Preventing dictionary attacks”. Computer,40(6): pp. 68–74, June 2007.

[28]. B. Pinkas and T. Sander. “Securing passwords againstdictionary attacks”. Proc. of 9th Conf. on Computer andCommunications Security, pp. 161–170, Nov. 2002