A Web Browser Extension based CAPTCHA By-passing technique

Mukund Kumar
Department of Computer Science and Engineering, Manav Rachna International Institute of Research & Studies

Shashank Kaushik
Department of Computer Science and Engineering, Manav Rachna International Institute of Research & Studies

Raja Siddharth Raju
Department of Computer Science and Engineering Manav Rachna International Institute of Research & Studies

Dr. Mukesh Kumar
Department of Computer Science and Engineering, Manav Rachna International Institute of Research & Studies

Dr. Prateek Jain
Accendere Knowledge Management Services, New Delhi

Abstract

The internet has been playing an increasingly important role in our daily life, with the availability of many web services such as email and search engines. However, these are often threatened by attacks from computer programs such as bots. CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) is a computer program which helps the server-side entity to determine whether the access is requested by a human or someone else. CAPTCHA is used so as to defend the automated programs, bots or hacker to use restricted areas of the areas and to prevent unauthorized access to the users account. CAPTCHA is widely used by various websites while a person tries to login as to prevent spams or any other harmful attacks. But CAPTCHA solving nowadays, has become an issue for humans. This paper will focus on CAPTCHA and its uses in addition to the study of various CAPTCHA breaking techniques that are proposed for a practical overview.

Keywords: CAPTCHA, Bypass, Automated Programs, Machine Learning, Bots

Introduction

With the growing world and exponentially increasing online transactional data, it is becoming important for a user to decide what data needs to be protected. Today's world is filled with enormous amount of technological advancements and innovations so preserving and disposing of data is to be done during the data management and planning phases. A captcha is basically a system program that is designed to differentiate human users from machine bots. In below Fig 1, a typical CAPTCHA has been shown.

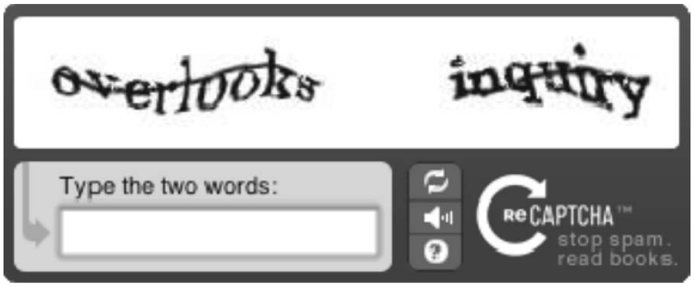


Fig 1: CAPTCHA

It is being implemented because Captcha's are generally used as a way to barricade the spams and automated extraction of data from websites. This form of user identification system has received many appreciations and criticism in the recent years. The users think that it might be a method of protecting our confidential data and avoid spams to an extent but they also believe that the captchas ultimately contribute more in delaying the work in progress. Captcha's needs to be re-entered if not recognized properly and may also lead to refilling the entire log-on page over and over again. Studies believes that an average person takes approximately a minimum of 10 seconds to solve a typical captcha [1]. There are 3 different types of captcha's that are presently used for the distinguishing purpose [2]

Text based captcha's: One of the simplest form of captcha's yet innovative for its initial developing stages. The queries asked by text-based captchas are not so difficult for a user to enter but may be really problematic for the bots to enter correctly.

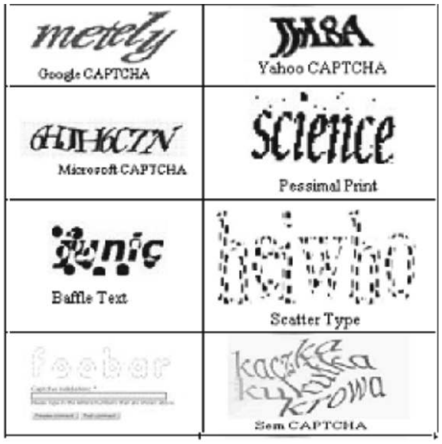


Fig 2: Text Based Captcha's

IMAGE based CAPTCHA's: Users identify images by performing the recognition test asked by the captcha. First image CAPTCHA which used named as ESP Pix and it was developed at Carnegie Mellon University.

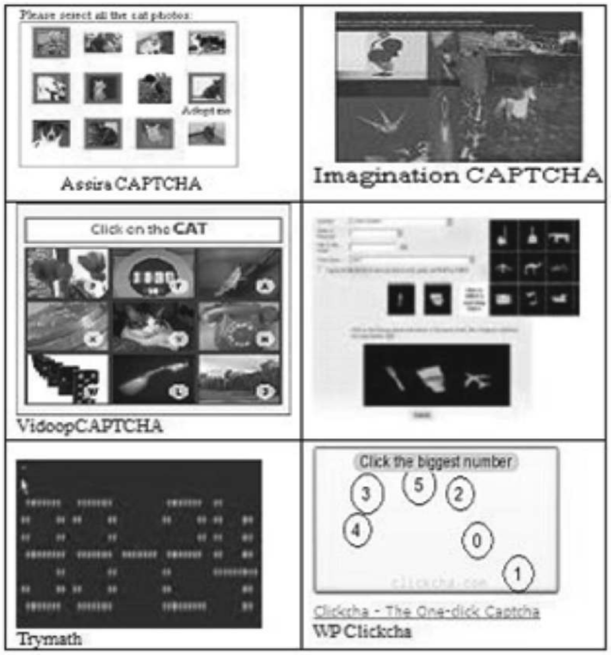


Fig 3: Image based CAPTCHA

AUDIO based **CAPTCHA's:** They basically rely on total human perception but making it impossible to solve by people having vision impairments. Specifically, in audio-based Captcha's text is synthesized and merged in with background noise, such as sound or unrecognized chatter [3].

VIDEO based **CAPTCHA's:** The user is given a moving object and is asked to perform a task.

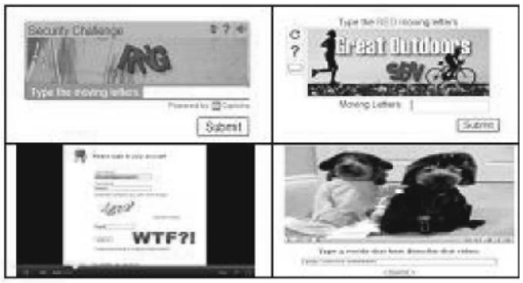


Fig 4: Audio/Video based CAPTCHA

II. APPLICATION OF CAPTCHA

a) Registering free via net forms: Millions of websites functioning on the internet offer free registering to the services such as the e-mail, online gaming's, social networking sites etc. Unfortunately, bots attack primarily these for personal benefits.

b) Online polling: An online survey where individuals participate and generate their responses via the net [4].

c) Web crawler: Basically, one of the computer scripts that browses the internet in an automated routine.

d) Phishing attack: A type of attack where users try to acquire usernames, bank details or other credentials by camouflaging as an authenticate user.

III. PROBLEMS FACED USING CATPCHA

Though CATPCHA are used for security reasons by the websites, but they sometimes become hectic for humans. It is proved that a human takes average time taken by a human to solve the CAPTCHA is 10s [5]. Some of the problems faced while solving the CAPTCHA are [6]:

reCaptcha errors: - reCaptchas often malfunction and cause trouble for the user. In this case, the user has difficulty in using the site.

Time constraint: - The average time taken by a human to solve CAPTCHA is 10 seconds. Considering the multiple captcha prompts faced by the user, it's very time constraining.

Mobile Optimization Errors: - Many sites are only optimized for computers, and their mobile users may face problems in trying to resolve captcha on their phones due to the broken CAPTCHA system.

ad UI element: - Captchas make for a bad user interface element on a website. They give horrendous UI experience and turn away visitors. Captcha is a Bad UI Element that inconveniences users.

IV. BREAKING THE CAPTCHA

Previous section mentioned some the problems that user face solving the CAPTCHA like time consuming, solving issues etc. Therefore, bypassing or breaking the CAPTCHA is the only solution to it. Till now, there is not a single full-fledged working model that could help us break the CAPTCHA. Therefore, this section aims to brief us about various technology that can be used to break/bypass the CAPTCHA.

Various authors around the world have proposed some CAPTCHA breaking techniques and some of them are: -

A. Break CAPTCHA using DeCaptcha

The author [8] suggested a method to break down the CAPTCHA image into characters so that it will become easy for the CAPTCHA solver to solve it. He analyzed various CAPTCHA like eBay, Baidu, Authorize, Blizzard, and ReCaptcha and tried to study different patterns these CAPTCHAS have. Then using C#, he prepared an extension to break down the CAPTCHA into characters which can be easily read by the CAPTCHA solver. Thus, the author broke down the process into 5 parts that are: -

Pre-Processing: - The first phase where the background of the CAPTCHA is removed and is represented in black and white form also known as binarized and then it is then stored into a form of binary values

Segmentation: - The stage where the DeCaptcha tool attempts to segment the CAPTCHA using the default segmentation technique known as CFS (Color Filling Segmentation) [7] algorithm which will allow the CAPTCHA to be segmented even if they are tilted.

Post-Segmentation: - In this stage, segments are processed individually so as to make the recognition simpler and efficient.

Recognition: - The stage where the tool recognizes the character which is used in the CAPTCHA. It recognizes what character it looks like after CAPTCHA is segmented.

Post-Processing: - This stage basically focusses on how to improve the obtained output is required.

Breaking CAPTCHA using Naïve Pattern Recognition

All the visual captcha strategies use a method of random shear distortion, which [8] is as follows: “the initial image of text is distorted by randomly shearing it both vertically and horizontally. That is, the pixels in each column of the image are translated up or down by an amount that varies randomly yet smoothly from one column to the next.

Then the same kind of translation is applied to each row of pixels (with a smaller amount of translation on average).” There are various captcha schemes, such as Word image, Random_letters_image, Number_puzzle_text_image. To attack the schemes, the basic attack algorithm can be used, which is as follows: Two separate colors are used for the background and the distorted challenge text. So as to easily separate the text and background. And only letters that were capital were used. Each letter in a captcha has a distinctive pixel count that's constant.

The basic attack is based on the above premises. And the vertical segmentation algorithm, works as follows:

Obtain the background pixel color to differentiate it from the foreground pixel color.

Map the image on a co-ordinate system and identify the segmentation line. There's a slicing process when a pixel not matching in background color is found.

This continues until it detects another vertical line not having the foreground pixels. This is the next line of segmentation.

Vertical slicing goes on from a pixel to the position that is right of the previous line of marked segmentation. But, the next line that is vertical doesn't have any foreground pixel isn't mandatorily the next line of segmentation. The line could be ignored on the count of being redundant. Henceforth, only when the process of vertical slice that's done cuts through the next letter, the next line that is vertical that won't have foreground pixels at all will be the next segmentation line.

Step #4 is iterated until the last segmentation line decided by the algorithm. So basically, a captcha bypass works like this.

Separate the text in the foreground from its background. It's easy to do this using an automated program. Then, a simple segmentation attack such as the vertical segmentation we discussed is implemented. Each character is made out using it's a unique pixel count by counting no. of pixels of foreground in each segment. Thus, once differentiated from the background. The foreground using segmentation are identified and thus broken.

Breaking CAPTCHA through OCR

This author [9] presented an idea to break the CAPTCHA using OCR (Optical Character Recognition). OCR is simply a digital form of texts, pictures or typewritten that are scanned [10]. OCR helps us to store the data in electronic form so that it can be easily searched, stored, processed and can be used for machine translation, text mining etc. Therefore, some of the open sources OCR are as follows:

GOCR: It is an OCR program that was advanced under GNU public license. The main feature of this program is that it scans the text image and converts it into text files and also can be used to translate barcodes. But it has some troubles related to serif fonts, strident images and handwritten content.

Tesseract OCR engine: Tesseract OCR engine is a free software developed by HP (Hewlett Packard) and sponsored by Google since 2016. This software is considered to be the most accurate software available that can read variety of formats plus with an additional feature of translation in over 60 different languages [11].

BREAKING TEXT-BASED CAPTCHA THOUGH VARIABLE WORD

There's a filter applied to detect disjointed areas in the image that has the captcha word. And it's stages, the steps of CAPTCHA breaking are as follows [12]:

CAPTCHA image acquiring and binarization. CAPTCHA word selection. Straightening CAPTCHA characters and word. This is the Word/character preprocessing stage.

Then now the character segmentation stage, which is as follows: Three-color bar encoding characters in CAPTCHA. Pruned skeletion generation. Character segmentation.

And lastly, the recognition stage, SVM classifier training and segmentation. The way for separating characters, segmenting them and their recognition.

There are different characters, the circular ones such as a, o, q etc are different px than others such as c, e, f, k, s, t z and others. So each characters px is pre-determined. Then a process of noise reduction happens on the basis of strict threshold. And other pattern recognitions is charted according to the areas different characters take. Then a skeleton is generated for the character selection. And we get the basis or backbone for the verification of those characters, what we start discoursed to us here is a pattern. Then there's segmentation based on color codes and letter skeletal patterns are matched to their respective identifiers. And after applying this algorithm, we get the characters. But as different captcha systems vary with distortion, foreground, background and pixels, so does the success rate of this method of breaking of captcha.

PROPOSED WORK/METHODOLOGY

After doing a brief analysis on breaking the CAPTCHA and different models suggested by different authors, we prepared just a suggestive model on how to bypass the CAPTCHA using a web browser extension. The steps are as follows:

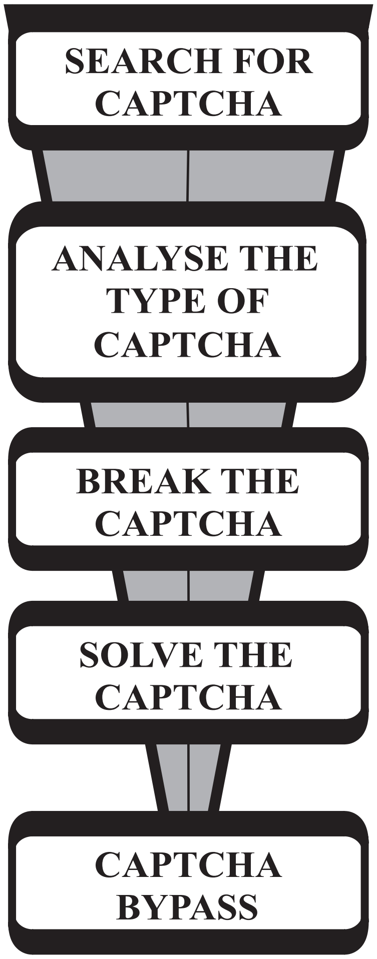
Search for CAPTCHA: - This step will make a run through on the web page that will have a CAPTCHA. User just need to enable the extension and click on the button to start the process.

Analyze the CAPTCHA: - This step will analyze the type of CAPTCHA that is used in the website. There will be two types, that are text CAPTCHA and image CAPTCHA

Break the CAPTHCA: - In this step, the CAPTCHA will be segmented into characters (if it is text) or pixels (if it is image) by using technologies mentioned in the above section.

Solve the CAPTCHA: - The main step where our actual program (actual code) will work. The automation where it will analyze our CAPTCHA and will solve it.

Fig 5: Simple Flow of CAPTCHA bypass using browser Extension



CONCLUSION

In this paper, we have discussed in brief about what CAPTCHA is, its implementation, benefits and application. The later part discusses about the problem of users they face while solving the CAPTCHA. The next part is the brief study of various techniques used in breaking the CAPTCHA. Thus, with the help of the above study, we therefore suggested a model to bypass the CAPTCHA using a web browser extension.

ACKNOWLEDGEMENT

We would like to sincerely bring our kind gratitude to Dr. Mukesh Kumar, Professor of Department of Computer Science and Engineering, Manav Rachna International Institute of Research and Studies, Dr. Prateek Jain, Accendere KMS Pvt. Ltd. for helping and guiding us in this paper formation.

REFERENCES

Bursztein, E., Bethard, S., Fabry, C., Mitchell, J. C., & Jurafsky, D. (n.d.). Retrieved March 30, 2018, from https://web.stanford.edu/~jurafsky/burszstein_2010_captcha.pdf.

E. Athanasopoulos and S. Antonatos. Enhanced captchas: Using animation to tell humans and computers apart. In IFIP International Federation for Information Processing, 2006.

L. von Ahn, M. Blum, J. Langford, “Telling Humans and Computers Apart Automatically Communications of the ACM”, vol. 47, no. 2, pp. 57-60, Feb. 2004.

Elie Bursztein, Steven Bethard, Celine Fabry, John C. Mitchell, Dan Jurafsky, How Good are Humans at Solving CAPTCHAs

https://en.wikipedia.org/wiki/CAPTCHA 7

https://www.thebalance.com/what-to-do-when-captchas-dont-work-897057 8

Bursztein, E., Martin, M., & Mitchell, J. (2011, October). Text-based CAPTCHA strengths and weaknesses. In Proceedings of the 18th ACM conference on Computer and communications security (pp. 125-138). ACM. 9

J. Yan and A.S. El Ahmad. A Low-cost Attack on a Microsoft CAPTCHA. In Proceedings of the 15th ACM conference on Computer and communications security, pages 543–554. ACM, 2008. 10

T Converse, “CAPTCHA generation as a web service”, Proc. of Second Int'l Workshop on Human Interactive Proofs (HIP'05), ed. by HS Baird and DP Lopresti, SpringerVerlag. LNCS 3517, Bethlehem, PA, USA, 2005. pp. 82-96. 11

Mori, Shunji, Hirobumi Nishida, and Hiromitsu Yamada. Optical character recognition. John Wiley & Sons, Inc., 1999. 12

Sharma, S., & Seth, N. (2015). Survey of Text CAPTCHA Techniques and Attacks. International Journal of Engineering Trends and Technology (IJETT), 22(6).

Tamang, Tsheten, and Pattarasinee Bhattarakosol. "Uncover impact factors of text-based CAPTCHA identification." In Computing and Convergence Technology (ICCCT), 2012 7th International Conference on, pp. 556-560. IEEE, 2012.

Starostenko, O., Cruz-Perez, C., Uceda-Ponga, F., & Alarcon-Aquino, V. (2015). Breaking text-based CAPTCHAs with variable word and character orientation. Pattern Recognition, 48(4), 1101-1112.