Best Practices For Choosing Non-Intrusive But Effective CAPTCHAs

Setia Budi [(1)]
sbudi@utas.edu.au

Abstract

Completely Automated Public Turing Test to Tell Computers and Humans Apart (CAPTCHA) is has become a part of computer security standard in order to distinguish humans from computers. This paper mainly discusses usability and robustness issues in CAPTCHA design and implementation. A brief introduction about CAPTCHA and some attempts to defeat CAPTCHA, including how to deal with it are also described in this paper.

Kata Kunci : CAPTCHA, Turing Test, web security

1. INTRODUCTION

In plain English, CAPTCHA can be defined as a computer program with capabilities to generate and to assess a set of tests that can distinguish humans from computers. In order to do this, CAPTCHA provides a set of tests that most humans can pass, however, the current computer programs cannot pass (Ahn, 2003).

According to Ahn (2004), the word CAPTCHA is an abbreviation for Completely Automated Public Turing Test to Tell Computers and Humans Apart. CAPTCHAs have similarities to Turing Tests which is a set of tests to distinguish humans from computers. In the original Turing Test, a human examiner or judge provides a set of questions to the two parties. One of the parties is a human and the other party is a computer, or more specifically an artificial intelligence system. Both parties act as a real human and try to answer each question asked by the examiner. At the end of the test, the examiner will tell which one of the parties is the real human. On the other hand, in CAPTCHA, the examiner that conducts the test is a computer system rather than a human. In CAPTCHA, a computer generates the test and assesses the answer. This is the reason why the word "Completely Automated" comes in CAPTCHA. Moreover, Ahn (2004) also mentions that CAPTCHA is not a Reverse Turing Test because the Reverse Turing Test refers to another test where two

[1] *School of Computing and Information System, University of Tasmania, Australia*

parties, one human and one computer, pretend to be a real computer. As in the original Turing Test, the human examiner in the Reverse Turing Test provides a set of questions to be answered by both parties, and based on the answers, the examiner will distinguish which one of the parties is the real computer. In other words, in the original Turing Test, both parties pretend to be a human. However, in the Reverse Turing Test, both parties pretend to be a computer. The examiner in both the original Turing Test and the Reverse Turing Test is a human. In contrast, the examiner in CAPTCHA is a computer. Therefore, CAPTCHA is more similar to the original Turing Test because the parties in both CAPTCHA and the original Turing Test pretend to be a real human.

Furthermore, Yan (2008) classifies CAPTCHA into three main categories, which are text-based schemes CAPTCHA, sound-based schemes CAPTCHA, and image-based schemes CAPTCHA. In text-based schemes CAPTCHA, the generated tests typically based on images of distorted characters which cannot be read by the current pattern recognition programs but they are still readable for most humans. In sound-based or audio-based schemes CAPTCHA, the user is asked to identify a distorted audio which current speech recognizer programs failed to recognize. In image-based schemes CAPTCHA, a set of distorted images is provided to the user to be identified.

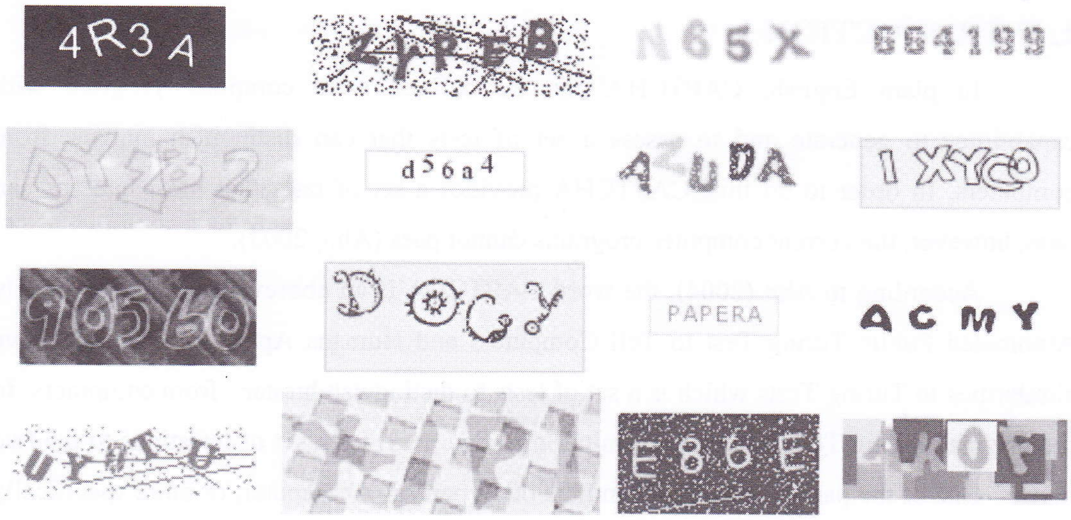


Figure 1. Text-based scheme CAPTCHAs

This research report will describe practical applications of CAPTCHA, the distortion methods that commonly used in CAPTCHA, the usability aspects in CAPTCHA, and several attempts to defeat CAPTCHA. It will then consider what is actually going to happen when a CAPTCHA is broken, and finally, a brief summary is provided to conclude the research report. For the purpose of further discussion and research, a list of references and bibliography are provided. Even though there are three different types of CAPTCHA, this research topic focuses on the text-based schemes CAPTCHA.

2. PRACTICAL APPLICATIONS

Text-based CAPTCHA is the scheme most widely deployed and implemented as a security solution (Yan, 2008). It provides an image consists of distorted alphabetical characters and asks the users to type and submit the valid original characters in order to prove that the users are real humans rather than computer programs (Ahn, 2008). This application is supported by the study conducted by Chellapilla (2008), which states that the current computer programs have difficulties in reading a distorted text even though it is still readable by the humans. Because of its capabilities to identify and to distinguish humans from computer programs, CAPTCHA is implemented on many different type of web sites to prevent automated computer programs from abusing the online services (Ahn, 2008).

Bots, automated computer programs created to abuse online services, have become a big issue for some web sites that provide online services. For example, in November 1999, there was an online poll released by slashdot.org asking about which is the best graduate school in computer science. In order to prevent a single voter from voting more than once, the IP addresses of each voter are recorded. Unfortunately, this kind of mechanism is not enough to provide a valid poll. Just after the online poll was released, the students from Carnegie Mellon University and the students from MIT started write an automated computer program to vote for their university. Students from both universities were competing each other to write a better bot in order to get the highest number of votes for their university in the online poll. At the end, the MIT got 21,156 votes, Carnegie Mellon University got 21,032 votes, and the other universities got less than 1,000 votes. Obviously, the result of the online poll is not valid and cannot be trusted since there was no guarantee that the voters are all humans (Ahn, 2004).

Another good example for CAPTCHA application is in free email services. Companies, such as Google, Yahoo!, Microsoft, and other companies that offer free email services have a big challenge to make sure that for every user that sign up for their email services is a human and not a bot. A bot can abuse the free email services by signing up for thousands of email accounts in every minute (Ahn, 2003). In order to overcome this problem, a CAPTCHA can be implemented as a part of user registration form for a new email account.

According to Pinkas (2002) and Namprempre (2007), CAPTCHA can also be implemented to improve the security level in an authentication process. One common threat in the authentication process is dictionary attack, where an attacker tries to guess the password for a targeted account based on the words that available in the dictionary. Since the password guessing process can be considered as a time consuming activity for a human, the attackers tend to write a computer program that can conduct an automated dictionary attack. Implementing CAPTCHA in an authentication form can prevent the automated dictionary attack programs from being able to iterate through the entire space of passwords.

3. DISTORTION METHODS

In order to distinguish humans from computers, text-based CAPTCHA implements four different kind of text distortion methods. The first method is translation, which basically tries to shift the characters either up or down, and left or right by an amount that is hard for a computer program to recognize. The second text distortion method is rotation. In this method, the characters are turned in either a clockwise or counterclockwise direction. Scaling is the other method which is used for text distortion. In this method, the characters are stretched or compressed either based on vertical direction (y-direction) or based on horizontal direction (x-direction). The last distortion method is warp. In this method, several different scales of elastic deformation process is applied to CAPTCHA images (Yan, 2008).

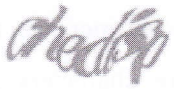
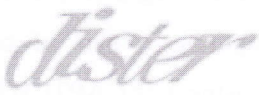
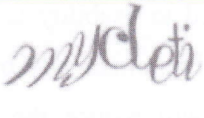
4. USABILITY AND ROBUSTNESS ISSUES IN CAPTCHA

According to Yan (2008), there are three main issues related to the usability and robustness in CAPTCHA: distortion related issues, content related issues, and presentation related issues. Distortion is a very essential component in CAPTCHA to ensure that the current computer programs cannot recognize the set of characters generated by the CAPTCHA. However, the method and the level of distortion being used also promotes several usability problems. The distortion in CAPTCHA eventually will reduce the readability of the characters. Many times, the distortion process itself produces ambiguous characters which are hard to be identified by the human users.

Yan (2008) identifies four common ambiguities in character pairs produced by CAPTCHA. The first ambiguity is between letter and number. Sometimes it is difficult to identify the distorted letter O from number 0, number 6 from letter G or b, number 5 from letter S or s, number 2 from letter Z or z, and number 1 from letter l or I. The second ambiguity is between number and number, such as distorted number 5 is hard to distinguish from number 6, number 7 from number 1, and number 8 sometimes looks similar to number 6 or 9. The third ambiguity is between letters and letter, or between letters and letters. In some distortion processes, the letters "vv" can be very similar to letter "W". Another similar kind of ambiguity also happen between letters "cl" and letter "d", letters "nn" and letter "m", letters "rn" an letter "m", letters "rm" and letters "nn", letters "cm" and letters "an", and many others. The fourth ambiguity is between letters and clutters. Clutters are introduced as a part of the distortion process in some CAPTCHA applications such as the CAPTCHA implemented by MSN. The clutters are used in order to increase the distortion level, and usually they appear in the form of random arcs. However, in many cases, they bring ambiguity to the human users in order to distinguish the arcs from the letter J or L, and from the number 7.

Considering these usability issues introduced by the distortion process, it is essential to choose carefully what kind of distortion methods are going to be used in CAPTCHA and in what level it is going to be implemented. It is good to implement a high level of distortion in order to promote the robustness of CAPTCHA from being defeated by any character recognizer programs. However, the CAPTCHA itself become useless when the generated characters cannot be read by the human users.

Table 1.
Confusing characters in Google CAPTCHA (Yan, 2008)

Image	Confusing characters
	Is the middle part 'd" or connected "cl"?
	Another case of "cl" or "d" confusion.
	Another case of "cl" or "d" confusion.

The second usability issues in CAPTCHA is related to the content. According to Yan (2008), there are four factors in content that can promote usability issue in CAPTCHA: the size of character set, the use of lexical and nonlexical string, the length of the string, and the use of offensive words. Using a larger size of character set will eventually improve the resistance to random guessing attacks. However, it also reduces the usability because a larger character set will produce a higher number of characters that look similar after the distortion process. In addition, Bursztein (2011) shows that in average, human can solve CAPTCHA tests using numeric as character set with the success rate of 98%. However, this number drops to 82% for CAPTCHA tests using alpha numeric as character set.

The second content related factor is the usage of lexical and nonlexical string. Obviously, CAPTCHA that using dictionary word (lexical) scheme has a higher usability degree compared to the one that using random string (nonlexical) scheme. There are two main reasons that support this argument. The first one is humans in general can type word faster compared to when they're typing a predefined random strings. The second one is, it is easier for human to identify a distorted word compare to recognize individual distorted characters. The drawback of using the dictionary word scheme in CAPTCHA is the vulnerability to dictionary attack. However, Yan (2008) argues that there is no problem in dictionary word scheme CAPTCHA, and the dictionary attack threat can be overcome by improving the segmentation resistance.

String length is another content related factor that can promote usability issues. The implication of string length in the usability is highly depend on the lexical or nonlexical string scheme that being used. If the non-lexical string scheme is in used, then the longer string length eventually will reduce the usability. This is happen because there are more distorted characters that required to be identified by the users. In contrast, the longer string length will increase the degree of usability when the lexical string scheme is being used. It is easier for the human users to identify a word when there are more characters available, since humans are good at inferring whole pictures from only partial information. Yan (2008) also mentions that by having a longer string, it can increase the CAPTCHA resistance to random guessing attack since there are more characters available to be recognized and to be guessed by the character recognizer programs. Furthermore, Bursztein (2011) suggests that by using random string length as a part of core principles in CAPTCHA design, it can effectively increase the robustness of CAPTCHA.

The other factor in CAPTCHA content related to usability issues is the usage of offensive words. Every offensive word that generated automatically by CAPTCHA has a negative effect in user satisfaction and eventually will reduce the usability. The best solution to overcome this problem is by keep maintaining the blacklist of words which are considered as offensive, and prevent it from being generated by CAPTCHA (Yan, 2008).

The last usability issues in CAPTCHA, according to Yan (2008), is related to presentation factors such as the use of multiple font type and multiple font size, and also the use of color and background pattern. Bursztein (2011) argues that using multiple font types and multiple font sizes in CAPTCHA is a good design principle to produce a better CAPTCHA since it can effectively decrease the success rate of the character recognizer programs to identified the original strings. Moreover, there is no negative impact to usability introduced by the use of either multiple font types or multiple font sizes. Furthermore, in order to increase the usability and the robustness of CAPTCHA, initially color and background pattern are introduced in CAPTCHA. The use of colors both in font and in background pattern was believed can increase the usability since color is appealing and can facilitate the humans to recognize the string. Moreover, it can also be used to protect the string from being recognized by the character recognizer programs. However, based on further studies, both Yan (2008) and Bursztein (2011) are agree that the use of color scheme does not promote any positive impact to the usability, on the contrary, it introduces more usability issues without any significant improvement in the robustness level.

5. ATTEMPTS TO DEFEAT CAPTCHA

Similar case to the other security protocols, there are also several attempts rise to defeat CAPTCHA. According to Yan (2008), the attempts to defeat CAPTCHA are mainly came from computer vision studies and also from document analysis and recognition communities. In 2003, Mori publishes an algorithm, which is based on a complex pattern recognition process, to defeat EZ-Gimpy CAPTCHA. The algorithm produces a significant success rate, nearly 92% (Mori, 2003). This tremendous achievement is followed by a new technique introduced by Moy in 2004. The algorithm introduced by Moy (2004) to defeat CAPTCHA is based on the distortion estimation technique and it is proven as an another effective way to defeat the EZ-Gimpy CAPTCHA with the success rate nearly 99%. Furthermore, Chellapilla (2004) shows a very interesting study about how the visual CAPTCHAs can be defeated using machine learning based algorithms with the success rate between 4.89% to 66.2%. In 2008, Yan introduces a new approach to defeat CAPTCHA that focusing more on the fatal design errors that found in every CAPTCHA implementation rather than on the complex computer vision or machine learning algorithms.

6. HOW SPAMMERS DEALING WITH CAPTCHA

Bajaj (2010) provides an interesting report related to how the spammers dealing with the CAPTCHA protected website. The report shows that in order to pass the CAPTCHA test, several spammers are willing to pay people in India, Bangladesh, China, and other developing countries to do the test. The payment rates are between 80 cent to 1.2 USD for every 1,000 CAPTCHA test. Even though the payment rate can be considered as low payment compared to the payment rate for common data entry jobs, but still it can attract many young people in the developing countries to take the job. By doing this job, unskilled male farm workers in India earn around 2 USD in a day. However, this kind of action cannot be considered as an attempt to defeat CAPTCHA, because the main function of CAPTCHA is to distinguish humans apart from computers. Moreover, Bajaj (2010) also mentions in the report that after some periods of time, the productivity of people that doing this job will decline since it is a monotonous job and gradually they are going to lose their interest.

7. A WIN WIN SOLUTION

Ahn (2004) argues that CAPTCHA promotes a win win solution for both the computer security field and the Artificial Intelligence (AI) field since CAPTCHA itself basically is a security protocol that brings an open problems from AI field, which is considered as hard to be solved by the AI community. This brings a mutual situation, either a CAPTCHA remains secure and it still can be used as a solution to distinguish humans from computers, or the CAPTCHA is broken and an open AI problem becomes solved. In every new attempt to defeat CAPTCHA, eventually it also bring a new improvement in the field of AI. Furthermore, Ahn (2004) mentions that CAPTCHA is how the lazy cryptographers doing AI, because CAPTCHA will attract malicious programmers to work on an open AI problem in order to break the security protocol.

Improvement in CAPTCHA can be achieved by bringing and adopting a new open problem from AI field which is still considered as hard to be solved by the AI community. In order to bring an AI problem to be useful as a basis to construct a security solution, it requires an automated way to generate the problem including the solution for that problem. Therefore not every AI problem can be implemented in CAPTCHA. The problem itself must be clearly and precisely defined. This is necessary in order to implement the AI problem in security because it gives the AI community a concrete goal to work on. This can guarantee that the AI problem which is used for security purposes can also bring a positive contribution for AI field (Ahn, 2003).

8. CONCLUSIONS

Usability and robustness are two fundamental issues with CAPTCHAs. Usability in text-based scheme CAPTCHA is related to the guarantee that the strings generated by CAPTCHA are still considered as human readable. On the other hand, robustness in text-based scheme CAPTCHA is related to the guarantee that the strings generated by CAPTCHA cannot be read by any character recognizer programs. It is essential to balance between the usability and the robustness in order to produce an effective CAPTCHA. Using longer lexical strings with random string length are proven to be good practices in order to promote effectiveness in CAPTCHA. On the contrary, using color and complex background pattern are proven to be ineffective since it can reduce the usability without any significant security level improvement. Moreover, every attempt to defeat CAPTCHA brings a new improvement to the field of AI, because CAPTCHA itself basically a security protocol that utilizes the open problems in AI which are considered as hard to be solved by the AI community.

Reference

Ahn, L.V. et al. (2003) CAPTCHA: Using Hard AI Problem for Security. *Proceedings of the 22nd international conference on Theory and applications of cryptographic techniques*, p.294-311.

Ahn, L.V. et al. (2004) Telling humans and computers apart automatically. *Communications of the ACM*, 47 (2), p.56-60.

Ahn, L.V. et al. (2008) reCAPTCHA: Human-Based Character Recognition via Web Security Measures. *Science Magazine*, Iss. 5895 p.1465-1468.

Bajaj, V. (2010) Spammers Pay Others to Answer Security Tests. *The New York Times*, [online] 25 April. Available at: http://www.nytimes.com/2010/04/26/technology/26captcha.html?_r=1&src=me&ref=technology [Accessed: 08 May 2012].

Bursztein, E. et al. (2011) Text-based CAPTCHA strengths and weaknesses. *Proceedings of the 18th ACM conference on Computer and communications security*, p.125-138.

Chellapilla, K. et al. (2008) Designing human friendly human interaction proofs (HIPs). *Proceedings of the SIGCHI conference on Human factors in computing systems*, p.711-720.

Chellapilla, K. and Simard, P. (2004) Using Machine Learning to Break Visual Human` Interaction Proofs.*Neural Information Processing Systems (NIPS)*, p.265-272.

Mori, G. and Malik, J. (2003) Recognizing objects in adversarial clutter: breaking a visual CAPTCHA. *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1 p.134-141.

Moy, G. et al. (2004) Distortion estimation techniques in solving visual CAPTCHAs. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2 p.23-28.

Namprempre, C. and Dailey, M. (2007) Mitigating Dictionary Attacks with Text-Graphics Character CAPTCHAs. *IEICE transactions on fundamentals of electronics, communications and computer sciences*, p.179-186.

Pinkas, B. and Sander, T. (2002) Securing passwords against dictionary attacks. *Proceedings of the 9th ACM conference on Computer and communications security*, p.161-170.

Yan, J. and El Ahmad, A. (2008) A low-cost attack on a Microsoft captcha. *Proceedings of the 15th ACM conference on Computer and communications security*, p.543-554.

Yan, J. and El Ahmad, A. (2008) Usability of CAPTCHAs or usability issues in CAPTCHA design.*Proceedings of the 4th symposium on Usable privacy and security*, p.44-52.

Bibliography

Gupta, M. (2008) *Handbook of Research on Social and Organizational Liabilities in Information Security* . Information Science Reference.

Scambray, J. et al. (2010) *HACKING EXPOSED WEB APPLICATIONS*. 3rd ed. McGraw-Hill Osborne Media.

Viega, J. (2009) *The Myths of Security: What the Computer Security Industry Doesn't Want You to Know*. O'Reilly Media.

Zittrain, J. (2009) *The Future of the Internet*. Yale University Press..