This paper has been published in *Computers Helping People with Special Needs: 14th International Conference, ICCHP 2014, Paris, France, July 9-11, 2014, Proceedings, Part I*, *Lecture Notes in Computer Science*, vol. 8547, pp 129-136, by Springer in 2014. The original publication is available at http://link.springer.com/chapter/10.1007/978-3-319-08596-8_19.

jCAPTCHA: Accessible Human Validation

Matthew Davidson[1] and Karen Renaud[2] and Shujun Li[3]

[1] BT, London, UK (matthew.davidson@bt.com)
[2] School of Computing Science, University of Glasgow, UK (karen.renaud@glasgow.ac.uk)
[3] Department of Computing, University of Surrey, UK (shujun.li@surrey.ac.uk)

Abstract. CAPTCHAs are a widely deployed mechanism for ensuring that a web site user is a human, and not a software agent. They ought to be relatively easy for a human to solve, but hard for software to interpret. Most CAPTCHAs are visual, and this marginalises users with visual impairments. A variety of audible CAPTCHAs have been trialled but these have not been very successful, largely because they are easily interpreted by automated tools and, at the same time, tend to be too challenging for the very humans they are supposed to verify. In this paper an alternative audio CAPTCHA, jCAPTCHA (Jumbled Words CAPTCHA), is presented. We report on the evaluation of jCAPTCHA by 272 human users, of whom 169 used screen readers, both in terms of usability and resistance to software interpretation.

1 Introduction

One of the blights on the web is the pervasiveness of automated software agents that masquerade as humans to attack websites [8]. To counteract this, the **C**ompletely **A**utomated **P**ublic **T**uring tests are used to tell **C**omputers and **H**umans **A**part (CAPTCHA) [16]. These are interactive tests that human users can pass but which are difficult for software attackers to solve. We will refer to this metric as the **E***asy* **4** **H***umans,* **H***ard* **4** **S***oftware* (E4H-H4S) test.

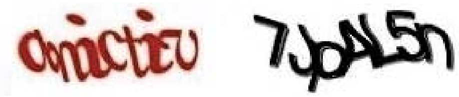

Fig. 1. Example of letter shape distortion CAPTCHAS (Google & Yahoo)

Generally CAPTCHAs are visual and consist of distorted text that users are required to decipher. Sometimes the shape and positioning of letters are changed (Figure 1) or background noise is added [12]. Such CAPTCHAs are often difficult to read or easy to break using specially designed crackers, failing the E4H-H4S test [4, 11].

Offering only visual CAPTCHAs ignores a sizeable portion of the online community. Hence audio CAPTCHAs have been introduced: users transcribe the characters that they hear, instead of those they see. Many have, thus far, failed the E4H-H4S test [13, 10].

The World Wide Web Consortium lists CAPTCHAs as one of the greatest security-related problems for users who "have low vision, or have a learning disability such as dyslexia" [2] and accommodating these users is a legal requirement in the UK [1]. Accordingly, we have developed a novel audio CAPTCHA that our user study shows is usable and accessible.

The rest of the paper is structured as follows. Section 2 reviews the current state of play with respect to audio CAPTCHAs. Section 3 then presents the jCAPTCHA solution, and explains how it was evaluated. Section 4 reflects on the results and Section 5 concludes.

2 Background

Suggested success rates for human users of CAPTCHAs should be around 90% and 0.01% for automated systems [5]. Achieving this is non-trivial. Bigham *et al.* [3] found that an audio CAPTCHA used to secure a high school course website was impenetrable by any of his 15 blind students. A 2008 study into the usability of the 8 digit audio CAPTCHA reported only a 46% pass rate. Based on a review of the literature, three particular design aspects are pertinent: *content, timing* and *accessibility.*

Content

Most audio CAPTCHAs require a user to hear, recognise and transcribe what they hear. As a first step, CAPTCHAs have to make sure that the articulated words are relatively common and easy to spell, then that the accent is easily understandable. In reality, most audio CAPTCHAs exclusively use digits to avoid spelling errors.

Similarly to visual CAPTCHAs, audio CAPTCHAs have to resist automated attacks. Often the digits are distorted, or background noise added, to resist automated recognition efforts. A popular resistance method is to add formatted human speech, perhaps played backwards or at a different volume to the characters of the actual CAPTCHA. This is supposed to make it harder for automated attacks to segment the digits. Unfortunately attackers have quickly found a way to strip this from the CAPTCHAs [6]. Moreover, composing audio CAPTCHA clips of digits, a very limited vocabulary, weakens the CAPTCHA unacceptably [15].

Using words increases the size of the vocabulary which makes CAPTCHAs harder to decipher automatically [15]. Language-based speech recognition tools have adapted by making use of contextual clues to ease the attacking process. They examine articulated words in context and identify a word by using both the audio characteristics of the word itself and the probability of such a word occurring, given the surrounding words. Contextual clues within phrases ease recognition for humans too [14].

Using words unfortunately re-introduces the spelling issue. One way of addressing this is to relax the exactness requirement. Non-exact matching will assist humans but not necessarily improve the success rates of automated attacking systems [15] so this is a technique that would be worth considering to improve E4H.

Timing:

A visual CAPTCHA is processed as a unit, whereas an audio CAPTCHA needs to be processed sequentially. The duration of audio CAPTCHA clips usually ranges from 3 to 25.1 seconds. Solving may well require multiple replays of the clip. The previously reported solve time is 65.64s [10]. Using radio clips to form the audio CAPTCHA could reduce the solve time to only 35.75s . Using words instead of nonsense characters also decreases the solve time since the context assists recognition [14, 10].

Accessibility

The final problem is related to the use of screen readers. CAPTCHAs require the user to enter an answer in a text box. Screen reader users are disadvantaged as they do not have access to their problem source whilst entering their answer, ergo, the answer must be entered from memory. Screen reader users have likened this to having visual CAPTCHAs with the problem image and answer text box on different webpages. Playing the audio clip also usually requires leaving the text entry field. The screen reader will normally narrate the page contents as the user navigates to the playback controls, and can then can talk over the audio CAPTCHA, inadvertently confusing the user.

3 jCAPTCHA Evaluation

In proposing a new kind of CAPTCHA we have attempted to address content and timing by using words derived from public media (to improve usability), and to design specifically to accommodate screen readers.

jCAPTCHA is an audio CAPTCHA that uses words as content *out of context*. As such, they rely on grammatical noise to fool language model-based speech recognition tools. The presence of grammatical noise avoids the need for further noise to be added to the audio clip. Therefore the answer can be ‘hidden in plain sight’ allowing humans to have a pleasant and straightforward experience in solving the jCAPTCHA but rendering current automated speech recognition tools unreliable.

The jCAPTCHAs were generated by manually concatenating audio clips from publicly available media to construct unusual phrases. The text used for the jCAPTCHAs can be viewed in Table 1.

ID	jCAPTCHA text	ID	jCAPTCHA text
1	very impressive helping hand	2	in britain vanilla look like
3	move on completely silent	4	into the water slightly forward
5	silent industry lift days	6	lift dinner push beauty
7	prize electric car ages	8	push food list guests
9	prize days screw push	10	bone fitness age glorious

Table 1. jCAPTCHA ID with the expected answer text

An evaluation webpage that allowed users to solve the ten jCAPTCHAs was implemented. High contrast colours were used and extraneous html elements removed to improve screen reader navigation and control. Bespoke user controls were offered to visually impaired users allowing them to transcribe jCAPTCHAs without needing to switch back and forth between media controls and the text entry field. Screen reader users use the 'full stop' key to play the audio file while the text field is active, and use the standard alphanumeric keys to enter their answer during or while the audio is playing.

The evaluation follows a design very similar to other studies assessing the usability of CAPTCHAs [10, 7, 13]. To accommodate human error or spelling mistakes, a Levenshtein distance [7] of two was allowed when judging the correctness of an answer. For the purpose of this experiment a distance of one was allowed for each word, with a possibility of one word being completely incorrect, whilst the answer as a whole still being judged as correct. Therefore in the answer to a 5-word jCAPTCHA 4 of the words must have at most a Levenshtein edit distance of one. The edit distance was chosen to allow for differences in pluralisation of words and typos, without accepting majorly different phrases. By evaluating, we wanted to answer the following questions:

- What is the success rate of jCAPTCHAs (i.e. can users comprehend them?)
- Is the experience more enjoyable for users compared to other audio CAPTCHAs?
- How long does it take to solve?
- Do the embedded controls ease the process?

The evaluation involved the following steps:

1. Demographic Questionnaire, to collect name, email address, age range, visual impairments, use of screen reader, IT expertise, hearing problems, other disabilities and spoken languages.
2. An initial training was given to allow participants to familiarise themselves with use of the site and solving jCAPTCHA.
3. Ten jCAPTCHAs were presented, one at a time. The users listened to an audio clip and attempted to type the words that they heard. Participants were then given the expected answer along with their own answer, as well as an indication of whether their answer was deemed close enough to the expected answer to have been accepted.

4. An exit questionnaire to gather satisfaction ratings and to collect additional comments.

After obtaining ethical approval, participants were recruited using social media and advertisements within the visually impaired community (talking newspapers and blind institutions). Respondents used the website with their traditional web-browsing set up, with or without screen readers.

3.1 Evaluating E4H (Easy 4 Humans)

272 individuals (173 Male, 96 Female, 3 Undisclosed) participated (138 aged between 18 and 30, 72 between 31 and 50 and 55 were 51 or over; 7 participants chose to not disclose their age range). 169 participants used a screen reader whereas 103 did not.

A number of user behaviours were monitored on the answer pages of the experiment. Time taken to submit an answer to each jCAPTCHA [4], key presses on input box and number of plays of jCAPTCHA audio were recorded. These measurements can be used to determine the process the user went through when typing their answer (multiple changes to words, misspellings, multiple plays, etc.).

Figure 2 shows the time taken to submit an answer for each jCAPTCHA with and without a screen reader. The mean time was 27.12 seconds. This measurement includes the time taken to load the page, listen to the audio at least once and to submit an answer. The graph also shows the success rate for users with and without a screen reader.

89 of the 103 non-screen reader users found jCAPTCHA easy to use. One non-screen reader user reported that they '*were frustrating to use, would rather use normal way of doing it*'. Luckily this was the only participant to express a purely negative opinion on the jCAPTCHA system.

139 of 169 screen reader users felt that jCAPTCHA was easy to use. Most users went on to say that the idea was much easier than alternatives. 22 of 169 users commented negatively about the clarity of the words in the audio clips. Common problems were the rate of the words in the audio or the pitch. A small number wanted the word segmentation improved, or wanted the speech rate slowed down. The general consensus from participants was mostly positive except for these issues.

3.2 Evaluating H4C (Hard 4 Computers)

Traditionally a paper that introduces a new security mechanism would include a summary of its resiliency against attacks. The robustness of the technique could be demonstrated by showing an expression of the entropy of possible solutions or by attacking the CAPTCHA. The results of a brief evaluation of freely available

[4] The time used in the calculation is measured server side. It is a measure of the time (in seconds) from the page being sent until the answer is received.

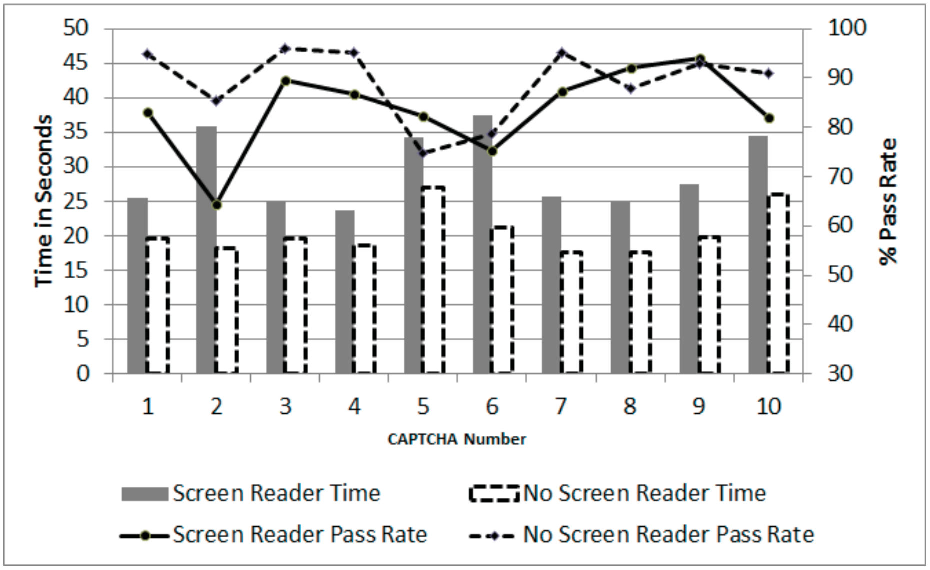

Fig. 2. Timings and Pass Rates

ASR tools is offered but these do not simulate a motivated CAPTCHA breaking attempt.

Two popular dictation programs (Dragon: Naturally Speaking[5] and iSpeech[6]) were used to test the resilience of jCAPTCHAs to automated software interpretation. The dictation software was first calibrated using clips of the same speaker for 31 words or phrases. The dictation software then attempted to interpret each jCAPTCHA. The responding transcription was manually recorded and evaluated for correctness using the same metric as used for participants.

Two of the 10 jCAPTCHAs were correctly interpreted: jCAPTCHA 4 was a perfect match. jCAPTCHA 5 was partially solved, with two words correctly identified and the Levenshtein edit distance [9] permitted the word 'gift' instead of the expected 'lift'.

jCAPTCHA 4 is, in hindsight, a suboptimal phrase since the word order could feasibly be used in normal conversation. It thus fails to meet the intended design goals. In the case of jCAPTCHA 5, the software submitted five words for the four word jCAPTCHA. Future heuristics for judging the correctness of the jCAPTCHAs should limit the number of possible words submitted in given answers, else multiple homonyms could be entered and used to break the jCAPTCHA.

[5] http://www.nuance.co.uk/dragon/index.htm
[6] www.ispeech.org

4 Discussion

jCAPTCHAs approach the desired 90% pass rate [5]. However, it should be acknowledged that these jCAPTCHAs only use one speaker, and the results may well differ if multiple speakers, accents and languages are employed in the audio clip formation stage.

The pass rate for all visually impaired users is 83.78% which is a remarkable improvement on the 46% pass rate given in [13]. It should be noted that Sauer's experiment [13] used a much smaller sample size (6) and ages were between 28 and 54, so the results may well be a worst-case scenario. The pass rate for reCAPTCHA is 70% [15] but for this study no demographic information was reported.

The mean answer submission time for screen reader users (31.46 seconds) is slightly faster than the reported time of 35.75 seconds in [10]. Only 10 screen reader users participated in the Lazar study whereas 169 participated in this study, making it difficult to conclude definitively which is the faster. Moreover, Lazar does not describe the method for recording answer times, it may only have measured time spent between the page having loaded and the answer message being sent to the server which would shorten the given time compared to the timings collected in our study. The mean answer submission time for non-screen reader users is 20.35 seconds, similar to the reported time of 22.8 seconds in [10]. It is likely that jCAPTCHAs do not offer significant improvements in answer submission time compared to radio clip based CAPTCHAs [10].

Before jCAPTCHA can be advanced as a viable alternative CAPTCHA a system needs to be created that can generate jCAPTCHAs automatically. To do this a corpus of audio clips and a reverse language model need to be created. The system should be tested with specially configured audio recognition toolkits to ensure that the words and ordering are resilient to more rigorous attacks.

5 Conclusion

The concept of a CAPTCHA was introduced and the accessibility issues explored. We then proposed a new, more accessible CAPTCHA called jCAPTCHA. We presented the results of our evaluation, which included participants using screen readers as well as those without visual impairments.

The results show that jCAPTCHAs are moderately resistant to recognition by off-the-shelf audio recognition programs. The jCAPTCHAs should be tested more rigorously with use of a toolkit that has been customised to recognise the types of clips used in the CAPTCHA. These jCAPTCHAs utilised only one speaker. Future jCAPTCHAs should use multiple speakers, accents and languages in order to diversify the vocabulary and thwart automated attacks.

The CAPTCHAs were created by hand, and, as such, it is not possible to give the entropy of all possible generated jCAPTCHAs. Given an automated system we would need to carefully consider how likely a training based attack can be developed. At this point we could then express the limits of the generation algorithm. If jCAPTCHA were ever deployed in the wild, it is predictable that attackers and defenders would enter a cat and mouse game of creating, updating and breaking the language models employed.

References

1. Equality act. http://www.legislation.gov.uk/ukpga/2010/15/contents. Accessed: 18/11/2013.
2. World Wide Web Consortium (W3C). Inaccessibility of CAPTCHA:2007. http://www.w3.org/TR/turingtest/. Accessed: 13/11/2013.
3. J. Bigham and A. Cavender. Evaluating existing audio captchas and an interface optimized for non-visual use. In *Proceedings of the 27th international conference on Human factors in computing systems*, pages 1829–1838. ACM, 2009.
4. E. Bursztein, S. Bethard, C. Fabry, J. C. Mitchell, and D. Jurafsky. How Good Are Humans at Solving CAPTCHAs? A Large Scale Evaluation. In *Proceedings of the 2010 IEEE Symposium on Security and Privacy*, pages 399–413, 2010.
5. K. Chellapilla, K. Larson, P. Simard, and M. Czerwinski. Designing human friendly human interaction proofs (HIPs). In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 711–720. ACM, 2005.
6. H. Gao, H. Liu, D. Yao, X. Liu, and U. Aickelin. An audio captcha to distinguish humans from computers. In *Electronic Commerce and Security (ISECS), 2010 Third International Symposium on*, pages 265–269. IEEE, 2010.
7. M. Gilleland. Levenshtein distance, in three flavors. *Merriam Park Software: http://www. merriampark. com/ld.htm*, 2009.
8. R. Gossweiler, M. Kamvar, and S. Baluja. What's up CAPTCHA? A CAPTCHA based on image orientation. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 841–850, New York, NY, USA, 2009. ACM.
9. W. J. Heeringa. *Measuring dialect pronunciation differences using Levenshtein distance*. PhD thesis, University Library Groningen, 2004.
10. J. Lazar, J. Feng, O. Adelegan, A. Giller, A. Hardsock, R. Horney, R. Jacob, E. Kosiba, G. Martin, M. Misterka, et al. Assessing the usability of the new radio clip-based human interaction proofs. In *Proceedings of ACM SOUPS Symposium On Usable Privacy and Security*, pages 1–2, 2010.
11. S. Li, S. A. H. Shah, M. A. U. Khan, S. A. Khayam, A.-R. Sadeghi, and R. Schmitz. Breaking e-Banking CAPTCHAs. In *Proceedings of 26th Annual Computer Security Applications Conference (ACSAC 2010)*, pages 171–180, 2010.
12. G. Mori and J. Malik. Recognizing objects in adversarial clutter: Breaking a visual CAPTCHA. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–134. IEEE, 2003.
13. G. Sauer, H. Hochheiser, J. Feng, and J. Lazar. Towards a universally usable CAPTCHA. In *Proc. of the 4th Symp. On Usable Privacy and Security (SOUPS'08), Pittsburgh, PA, USA*, 2008.
14. A. Schlaikjer. A dual-use speech CAPTCHA: Aiding visually impaired web users while providing transcriptions of audio streams. *LTI-CMU Technical Report*, pages 07–014, 2007.
15. J. Tam, J. Simsa, S. Hyde, and L. V. Ahn. Breaking audio CAPTCHAs. In *Advances in Neural Information Processing Systems*, pages 1625–1632, 2008.
16. L. Von Ahn, M. Blum, N. Hopper, and J. Langford. CAPTCHA: Using hard AI problems for security. *Advances in Cryptology EUROCRYPT 2003*, pages 646–646, 2003.