Improving Audio CAPTCHAs

Jennifer Tam, Jiri Simsa, David Huggins-Daines, Luis von Ahn, and Manuel Blum
Computer Science Department, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213
{jdtam, jsimsa, dhuggins, biglou, mblum}@cs.cmu.edu

ABSTRACT
CAPTCHAs are computer generated tests that humans can pass but current computer systems cannot. CAPTCHAs provide a method for automatically distinguishing a human from a computer program, and therefore can protect web services from bots. Most CAPTCHAs consist of distorted images, usually text, for which a user must provide some description. Unfortunately, visual CAPTCHAs limit access to the millions of visually impaired people using the web. Audio CAPTCHAs were created to solve this accessibility issue; however, the security of audio CAPTCHAs was never formally tested. We analyze the security of current audio CAPTCHAs, and provide a description and analysis of a new and improved audio CAPTCHA.

Categories and Subject Descriptors
K.4.2 [**Social Issues]:** *Assistive technologies for persons with disabilities.*

General Terms
Online Security, Abuse Prevention.

Keywords
CAPTCHA, Accessibility, Speech Recognition

1. INTRODUCTION
CAPTCHAs [1] are automated tests designed to tell computers and humans apart by presenting users with a problem that humans can solve but current computer programs cannot. Because CAPTCHAs can distinguish between humans and computers with a high probability, they are used for many different applications. CAPTCHAs prevent bots from voting continuously in online polls, automatically registering for millions of spam email accounts, automatically purchasing tickets to buy out an event, etc. Once a CAPTCHA is broken (i.e. computer programs can successfully pass the test), bots can impersonate humans and gain access to services that they should not, therefore, it is important for CAPTCHAs to be secure.

To pass the typical visual CAPTCHA, a user must correctly retype the text found in an image of distorted text. Because visually impaired users cannot see this type of CAPTCHA, audio CAPTCHAs were created. Typical audio CAPTCHAs consist of several speakers saying digits spoken at randomly spaced intervals. A user must correctly identify the digits spoken in the audio file to pass the CAPTCHA. To make this test difficult for current computer systems, specifically automatic speech recognition (ASR) programs, background noise is injected into the audio files. Although the noise makes the CAPTCHAs more difficult for computers to pass, it also makes the CAPTCHAs more difficult for humans to pass.

Since no official evaluation of existing audio CAPTCHAs has been reported, we tested the security of the audio CAPTCHA used by many popular Websites by running experiments designed to break it. Because we were able to break this audio CAPTCHA, we are creating a new audio CAPTCHA that is more secure, and easier for humans to pass. Section 3 includes more details regarding the design of the new audio CAPTCHA.

2. Assessment of Current Audio CAPTCHAs
Since automated programs can attempt to pass a CAPTCHA repeatedly, a CAPTCHA is essentially broken when a program can pass it more than roughly 5% of the time. The audio CAPTCHAs we tested consisted of digits spoken by random people, plus other human voices playing throughout as "noise." While many academic publications have attempted to break visual CAPTCHAs, to the best of our knowledge, none have investigated the security of audio-based CAPTCHAs.

To break the audio CAPTCHAs, we derive features from the CAPTCHA audio and use several machine learning techniques to perform ASR on segments of the CAPTCHA. There are many popular techniques for extracting features from speech. The three techniques we use are mel-frequency cepstral coefficients (MFCC), perceptual linear prediction (PLP) and relative spectral transform-PLP (RASTA-PLP). MFCC is one of the most popular speech feature representations used. Similar to a fast Fourier transform (FFT), MFCC transforms an audio file into frequency bands, but (unlike FFT) MFCC uses mel-frequency bands, which are better for approximating the range of frequencies humans hear. PLP was designed to extract speaker-independent features from speech [2]. Therefore, by using PLP and a variant such as RASTA-PLP, we were able to train our classifiers to recognize digits independently of who spoke them. Since many different people recorded the digits used by the audio CAPTCHAs, PLP and RASTA-PLP were needed to extract the features that were most useful for solving them.

In [2,3], the authors conducted experiments on recognizing isolated digits in the presence of noise using both PLP and RASTA-PLP. However, the noise used consisted of telephone or microphone static caused by recording in different locations. The audio CAPTCHAs we used contain this type of noise, as well as the added vocal noise, which is supposed to make the automated recognition process much harder.

Our approach to breaking the audio CAPTCHAs began by first training our classifiers on features generated from automatically segmented and labeled CAPTCHAs. We gathered 1,000 audio CAPTCHAs annotated with information regarding digit locations.

We randomly selected 900 of those CAPTCHAs to use for training, with the remaining 100 used for testing the accuracy of our classifiers and our ability to automatically solve CAPTCHAs. We ignored the annotated information from the 100 samples we used for testing. Our method for solving CAPTCHAs iteratively extracts an audio segment from a CAPTCHA, inputs the segment to one of our digit recognizers, and outputs the label for that segment. We continue this process until eight segments are labeled as digits or there are no unlabeled segments left. A segment to be classified is identified by taking the neighborhood of the highest energy peak of a yet unlabeled part of the CAPTCHA.

Once a prediction of the solution to the CAPTCHA is computed, it is compared to the true solution. Given the acceptance conditions of the audio CAPTCHA we analyzed, a prediction is considered valid if it meets any of the following criteria:

1) The prediction matches the true solution exactly.
2) Inserting one digit to the prediction would make it match the solution exactly.
3) Replacing one digit in the prediction would make it match the solution exactly.
4) Removing one digit of the prediction would make it match the solution exactly.

Our method will never produce a guess with nine digits, and therefore case 4 is irrelevant to our approach. Also outputs from our experiments show that even though theoretically our method could output seven or less digits, in practice it always outputs eight. This renders case 2 irrelevant to our approach as well. Nevertheless, case 3 applies quite often and helps us to achieve a better CAPTCHA solving accuracy. The results of our experiments (see Table 1) show that our method can solve 58% of the challenges.

3. Current Work: Developing a More Robust Audio CAPTCHA

Upon successfully breaking the current audio CAPTCHA, our objective now is to create a new audio CAPTCHA that achieves the following goals: (1) it cannot be broken by current ASR systems and (2) the human pass rate is at least 70%. To achieve the first goal, we plan to only use audio that has been analyzed by an ASR system which has produced poor results. To improve the human pass rate, we plan to take advantage of the human mind's ability to understand distorted audio through context clues. By listening to a phrase instead of to random isolated words, humans can decipher distorted utterances because they are familiar with the phrase, or they can use contextual clues to decipher the distorted audio. Using this idea, the audio for the new audio CAPTCHA will be taken from old-time radio programs in which the poor quality of the audio makes it difficult for ASR systems to transcribe. We expect the new audio CAPTCHA will be more secure than the current version and easier for humans to pass.

Symposium On Usable Privacy and Security (SOUPS) 2008, July 23-25, 2008, Pittsburgh, PA, USA.

Table 1. Test accuracies of (a) AdaBoost (b) SVM (c) *k*-NN. The first column includes the passing rate and the percentage of exact matches (in parenthesis). The second column includes digit recognition accuracy for fixed sized segments generated from test samples.

(a)

Feature type	CAPTCHA Test	Digit Test
MFCC	18% (6%)	74.60%
PLP SPEC	27% (10%)	82.40%
PLP CEPS	23% (10%)	77.60%
RASTA-PLP SPEC	9% (3%)	68.00%
RASTA-PLP CEPS	9% (3%)	68.00%
# of objects	100 CAPTCHAs	1008 segments

(b)

Feature type	CAPTCHA Test	Digit Test
MFCC	56% (43%)	94.80%
PLP SPEC	58% (39%)	96.80%
PLP CEPS	56% (45%)	95.40%
RASTA-PLP SPEC	36% (18%)	86.20%
RASTA-PLP CEPS	46% (30%)	89.90%
# of objects	100 CAPTCHAs	1008 segments

(c)

Feature type	CAPTCHA Test	Digit Test
MFCC	22% (11%)	81.30%
PLP SPEC	43% (25%)	93.80%
PLP CEPS	29% (14%)	79.10%
RASTA-PLP SPEC	24% (4%)	79.90%
RASTA-PLP CEPS	32% (12%)	83.30%
# of objects	100 CAPTCHAs	1008 segments

4. ACKNOWLEDGMENTS

We thank Roni Rosenfeld and Alex Rudnicky for discussing current problems in speech recognition with us to help us design the more robust audio CAPTCHA.

5. REFERENCES

[1] L. von Ahn, M. Blum, and J. Langford. "Telling Humans and Computers Apart Automatically," *Communications of the ACM*, vol. 47, no. 2 pp. 57--60, February 2004.

[2] H. Hermansky, "Perceptual Linear Predictive (PLP) Analysis of Speech," *J. Acoust. Soc. Am.,* vol. 87, no. 4, pp. 1738-1752, Apr. 1990.

[3] H. Hermansky, N. Morgan, A. Bayya, and P. Kohn, "RASTA-PLP Speech Analysis Technique," In *Proc. IEEE Int'l Conf. Acoustics, Speech & Signal Processing,* vol. 1, pp. 121-124, San Francisco, 1992.