Article can be accessed online at http://www.publishingindia.com

O-CAPTCHA - An Orientation Puzzle

Ramsha Rizwan*, Waqas Malik*, Asif Hasnain*

Abstract

Malicious applications or bots massively contaminate the authenticity, and seriously damage the integrity of online valuable resources. CAPTCHAs prove to be excellent tool for protecting precious online resources. The reason for the success of CAPTCHAs is their simplicity in usability. CAPTCHAs schemes are evaluated by two characteristics i.e. usability and robustness. This paper presents a novel CAPTCHA scheme called 'O-CAPTCHA' which provides user with a text based orientation challenge. It is easier for human being to judge the proper shape of letters/alphabets so they can complete the challenge quickly. The Proposed method is tested on both Smartphone devices and desktop computers. Extensive experimentation is performed on the proposed method in terms of usability analysis and invincibility of the scheme against automated Bot attacks. Proposed orientation layer can be incorporated in any modern text based CAPTCHA scheme making them stealthier and a tough nut to crack for the bots without compromising their usability.

General Terms: Security, Usable Security, Turing Test, Bots

Keywords: CAPTCHA, Robustness, Usability

1. INTRODUCTION

CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) are used for protecting online registration on different websites and online communities. CAPTCHAs can either be images, texts or audio clips. In text based CAPTCHAs, users have to enter the text displayed in a distorted image, whereas some websites provide audio CAPTCHAs along with text CAPTCHAs as an accessibility aid for visually impaired users, as discussed in (Bursztein E., 2009; Schlaikjer A., 2007).

CAPTCHA is a test which must be easy for humans to solve and difficult for bots to crack. ACAPTCHA scheme is considered to be a failure if it fails in any of these two aspects. These two aspects are known as usability and security respectively. CAPTCHAs have evolved into many types and shapes as discussed in (Bairda, H. 2005; Chow R., 2008; Desai A., 2009; Elson J., 2007; Gossweiler R., Kamvar M., 2009; Matthews P., 2010; and Shah N. A. 2009) but still most widely used ones are text based CAPTCHAs. The reason for the success of text-based CAPTCHAs is its simplicity and easy data set generation.

Figure 1.1 shows the first CAPTCHA scheme used by AltaVista as discussedin (Henry S., 2002). Bots failed to read the text from image as they were not equipped with image processing at that time. After that bots were designed to read text from images like OCRs which could read text from within the image. Initially CAPTCHAS were known as HIPs (Human Interaction Proofs) but after in (Ahnl. V., 2003) were given the name CAPTCHAs and since then they are known as CAPTCHAs.

Figure 1.1 First CAPTCHA Used by Altavista

2. RELATED WORK

CAPTCHAs have come in many shapes and forms in the last decade. In reCAPTCHA (Ahn 2008), the user is presented two words to read and enter in the text box. These two words are taken from a random phrase of an old book. These scans of old books are not readable by machines; humans on the other hand can read them easily. At the backend, server which sends this CAPTCHA to the users knows answer to one of the words and if user

* National University of Computer and Emerging Sciences, FAST, Islamabad, Pakistan. Email: ramsha.rizwan@live.com
* National University of Computer and Emerging Sciences, FAST, Islamabad, Pakistan. Email: malekwaqas@gmail.com
* National University of Computer and Emerging Sciences, FAST, Islamabad, Pakistan. Email: asifhasnain85@gmail.com

enters that word correctly server assumes that user has entered the second world correctly as well. The server will save that input from user as the legitimate spelling of the second word in its database to use in the future. Due to these innovations this reCAPTCHA is in use in many websites to protect their services Figure 1.1 shows the reCAPTCHA challenge.Such use of textual CAPTCHAs has proved very beneficial in digitizing great number of books. But on the other side reCAPTCHA still uses character deformation and added noise to confuse the Bots which ends up confusing humans as well.

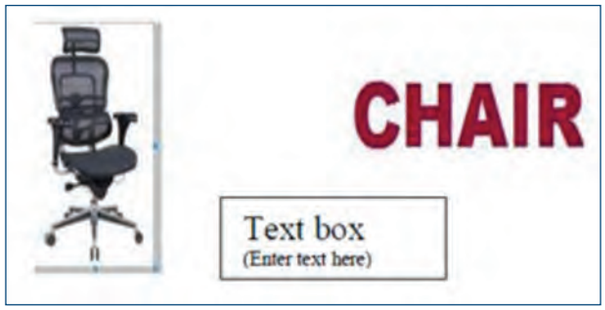
Figure 2.1 reCAPTCHA Sample

In (Shirali-Shahreza M., 2006), a Persian script CAPTCHA is discussed. Example of this CAPTCHA is shown in Figure 2.2. The CAPTCHA contained Persian text with no noise in the background there is no added text distortion. This CAPTCHA has a lot of use in areas of middle-eastern region where people understand this script and it is beneficial for that part of middle-eastern population which cannot read English text. Similarly for Hindi language a Devanagari script CAPTCHA has been discussed in (Yalamanchili, s., 2011).

Figure 2.2 Persian/Arabic Script CAPTCHA Challenge

In (Soni R., 2010), a new CAPTCHA is discussed which uses both picture and text. It is a very secure challenge where user is asked to firstly identify object and then select an appropriate name for it from a list. Then its name is typed in the textbox. Example is shown in figure 2.3 shows that a user has selected an image of chair and then his challenge is to type word ”CHAIR” in the text box.

Figure 2.3 Improved CAPTCHA



In (Kluever K., 2008), a video CAPTCHA challenge is discussed, that challenges user to tag the video after watching it which is compared to original tags specified by the video uploading person. The tag entered by the user is then compared with the tags of the video. The videos used in this case were from www.youtube.com as shown in figure 2.4 the user is being asked “Type 3 words that best describe this video”.

Watching a video consumes time, some time is consumed in buffering and video has its own runtime as well. This time is added to the time it takes to complete the challenge presented to the users.

Figure 2.4 Video CAPTCHA

In (Faymonville P., 2009) an open labeling CAPTCHA platform for computer vision researchers as shown in figure 2.5 is discussed, they showed one of the experiment images with an overlay of all user supplied boxes and the ground truth box. The average overlap of the user- supplied

bounding box with the ground truth box was 82.71%. They explored usability issues and security analysis for two different tasks of annotation and detection. They concluded with system sustainability issues in context of broader ecosystem for platform.

Figure 2.5 CAPTCHA-based Image Labeling on the Soylent Grid

In (Nazir M., 2011) an automated evaluation system with a name Captchaeker is discussed which calculates the strength of CAPTCHA quantitatively. It will help CAPTCHA designers to automatically check how secure and utilizable their proposed CAPTCHA scheme is, and how it can be improved. They showed that Captchaeker can predict hardness of a CAPTCHA in the testing set with accuracy over 80%, thus automatically judge how usable and secure a CAPTCHA is.

All the non-Textual CAPTCHA challenges discussed earlier did not prove to be long lasting success because the internet users are now very use to this idea of reading a text from and image and type in the text box, but for non-text based CAPTCHAs the users have to be first trained on the new scheme.

3. CAPTCHA BREAKING ATTACKS

Many forms of CAPTCHA breaking attacks exist.They vary from each other as they tackle different features of a CAPTCHA. Studying how attacks work can give vital insights which can suggest some features of a CAPTCHA that contribute positively towards robustness. Particularly in text based CAPTCHA schemes there are more or less four types of attacks

- Noise removal attacks
- Segmentation attacks
- Character recognition attacks
- Dictionary attacks

3.1 Noise Removal Attacks

Noise removal attack removes noise in the CAPTCHA image which has been added to confuse the Bots. Noise attacks are only used against those CAPTCHA schemes which contain added noise. Such a noise removal attack has been discussed in (Yan J., 2008) to break the Microsoft CAPTCHAs. Similarly in (Chellapilla K., 2005) preprocessing before segmentation is discussed which also resides in the noise removal area of CAPTCHA breaking attacks.

3.2 Segmentation Attacks

Segmentation attack is the next step in CAPTCHA breaking process after noise removal/preprocessing. Segmentation attack's objective is to separate each character present in the CAPTCHA image from the other. As most modern textual CAPTCHA contain characters joined together and sometimes even overlapped separating them from one and other is important in recognizing them. Such segmentation attacks have been discussed in (Ahmad el. a., 2012; Huang S., 2010; Yan J. 2008)

3.3 Character Recognition (OCR) Attacks

Recognition attacks come after segmentation step; it takes the segmented characters provided by segmentation attack and recognizes those using different Machine learning techniques and other recognition techniques include pixel count. OCR attacks have been discussed in (Li S., 2010; Simard, 2003) and pixel count recognition attacks have been discussed in (Yan J. 2007).

3.4 Dictionary Attacks

Dictionary attack takes the input of recognition attack and compares it with the word in dictionary (database of proper English words) and if word matches any word in dictionary, it confirms the authenticity of the recognition attack. However if no matches are found the nearest match is considered to be the word in CAPTCHA therefore correcting recognition Attack's output. Dictionary attacks can only work with only those CAPTCHA schemes in which proper Language words have been used. Dictionary attack will be useless if the characters of a CAPTCHA are random letters. Such dictionary attacks have been discussed in (Bursztein, E., 2011).

4. PROPOSED APROACH

The proposed CAPTCHA is O-CAPTCHA, where O stands for orientation. An extra orientation layer is being introduced between user and textual CAPTCHA. This is being done to maintain the usability of the original CAPTCHA but increasing the robustness. The new layer between the CAPTCHA and human is an orientation challenge, the text CAPTCHA itself is broken into number of sub-images and disoriented. The challenge for the human is to put the sub-images in-to proper orientation and then read it. Humans are good at perceiving details (gestalt perception) as described in (Ahmad A. el., 2010). The disoriented CAPTCHA is bit complex in the beginning but humans can make good sense of it when it is used in proper orientation.

4.1 Orientation Layer

O-CAPTCHAs are basically simple text based CAPTCHAs with an extra orientation layer. This layer challenges the user to put the disoriented and broken characters into its proper shape. Orientation alone will not be suffice to provide effective robustness as disoriented charter can also be recognized easily by automated attacks . In order to overcome this, characters are not only disoriented but also broken. This way Bot gets the wrongly segmented characters from O-CAPTCHA and fails to recognize it. This layer ensures that no noise or confusing deformation of text is need, thus making the CAPTCHA easy for humans to read. This way the orientation layer is giving user flexible usability i.e. by giving the user to fix the characters he has to read.

4.2 Text CAPTCHA Layer

Underneath the orientation layer there is a simple text based CAPTCHA. Any dataset can be used in this layer. In this paper we have used the dataset of Mega uplaod CAPTCHAs for testing purpose discussed in (Ahmada.el., 2010). Dataset was collected from source website (www.megaupload.com). Mega upload CAPTCHA scheme has high usability because it has neither added noise nor deformation of characters. The characters are in block capital letters and each character touches the other to avoid segmentation. Because of its good usability this dataset has been selected to be used in O-CAPTCHA.

4.3 User's Tasks

The user has to perform three tasks Adjust, Read and Type. User is given a puzzle with disoriented characters and scrollbar as shown in figure 4.1 that control the orientation of the CAPTCHA characters. As a first task user has to adjust the scrollbar with mouse (in PC environment) or with fingertips (In touch-screen environment) to put the characters in proper orientation. Second task is to read the characters and finally type them into the textbox like a simple text based CAPTCHA.

Figure 4.1. O-CAPTCHA Challenge

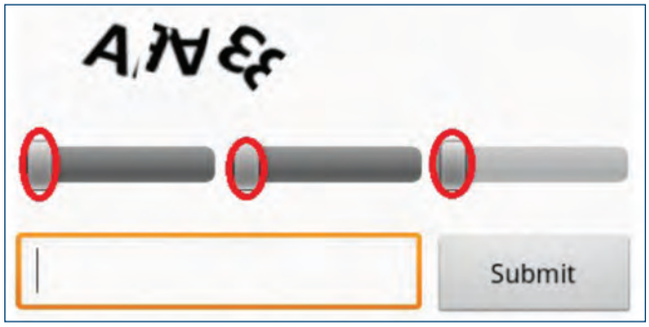
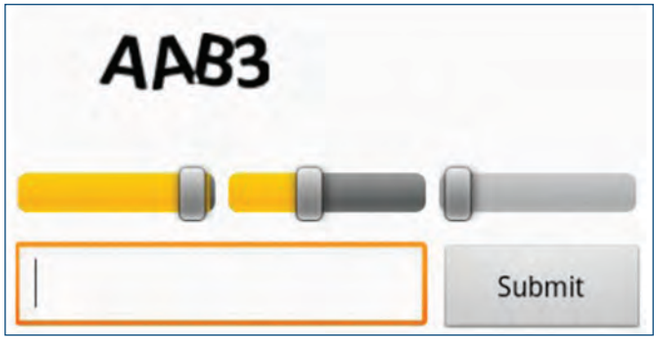


Figure 4.2 User Interface of O-CAPTCHA

Figure 4.3 After Putting the Characters in Proper Orientation



The interface used for the rotation is scrollbars; each pair is rotated by one scroll bar. Moving one scroll bar rotates

the pair of sub-images. The scrollbar is widely used interfacein different applications and thus its functionality can be used as an interface for humans in both computers and smart-phones.

The interface of O-CAPTCHA has been shown in the figure 4.2, moving the scroll bars rotates a pair of sub-images and the user has to put them in their proper orientation. Making pairs of sub-images improves the overall usability as the task is done is half time.

Figure 4.3 shows the proper orientation achieved by the user by adjusting the scroll bars to the right position. In this way the CAPTCHA is easily readable.

5. USABILITY ANALYSIS OF O-CAPTCHA

A CAPTCHA is said to have a high usability if it can be solved easily by humans. The usability of a CAPTCHA can only be determined by human testing for this purpose various experiments were carried out with human beings who were given the challenge to solve the O-CAPTCHA.

5.1 Experimental Setup for Usability TEST

For the usability test of new orientation CAPTCHA, people from different domains have been asked to test it. The application was tested by people on two different devices i.e. touch screen smartphone and a desktop/ laptop computer. There were no special requirements for computers; any ordinary desktop or laptop with internet connectivity would work. We used android smartphones for testing. The humans solved the challenge and their time to solve the challenge and ratings about their experience was examined. The sample of survey forms given to users are shown in appendix A.

5.2 Usability Experiment Results

This graph in figure 5.1 shows the effect on usability at different rotation step of scrollbar. Rotation angle step of the scrollbar means degree of rotation on sub-image on each movement of scrollbar. This graph shows that larger rotation step is giving good usability result. The ideal range of degree of rotation is determined from the values between 4 & 6 whereas values greater than 6 degrees makes the CAPTCHA less robust and values less than 4 makes it less usable. Thus from our experimental findings optimum range was found between 4-6 degrees.

The graph in figure 5.2 shows the effect on time to solve the CAPTCHA challenge by the number of sub-images it contains. Results showed that less number of sub-images give users ease to solve the challenge. The increase in the number of sub-division of the CAPTCHA results in decrease in the usability.

Figure 5.1 Rotation Angle vs. Time

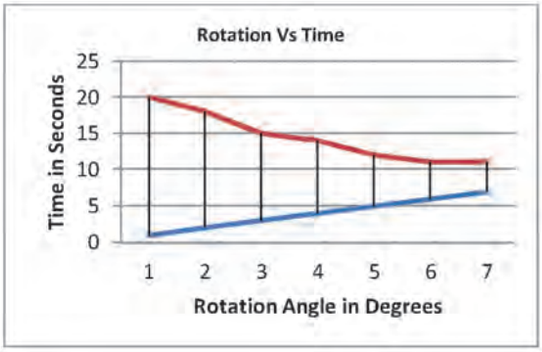
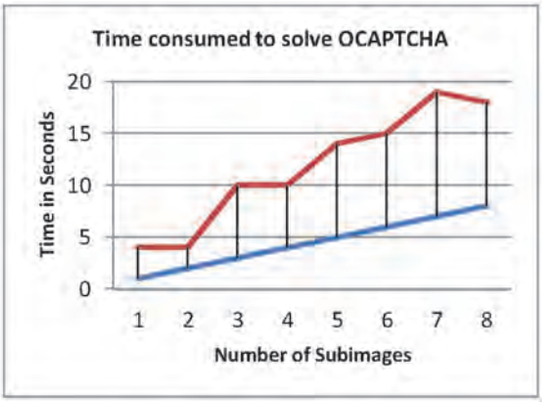
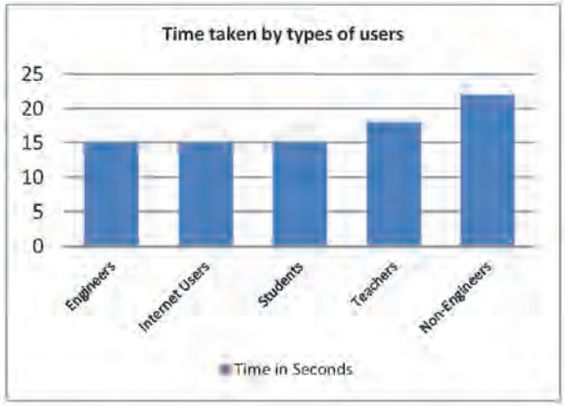


Figure 5.2 No. of Sub-images vs. Time



The graph in figure 5.3 shows how much usability curve is affected in different sections of society.

Figure 5.3 Usability in Different Type of Users



People from different backgrounds experience the usability in a different way which can identify outlier entities. In this case we can see the non-engineers take maximum time to solve the CAPTCHA challenge. Maximum time taken by non-engineering/ non-computer science background was 22 seconds.

This graph in figure 5.4 is showing the usability experience in computers and smart phones. Results show that the user gets a better usability experience in the Smartphone touch screen environment. In touch screen it's easier for the user to interact with the interface so they consume less amount of time as compared to the user on computer using mouse.

Figure 5.4 Usability in Different Device Types

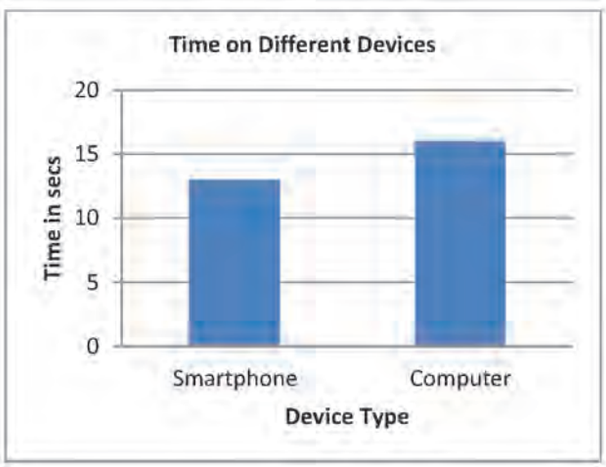
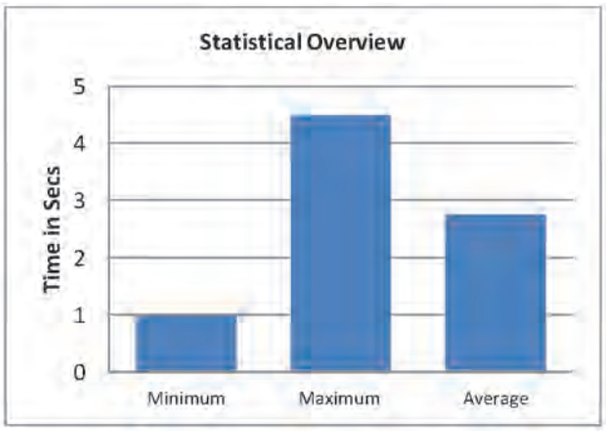


Figure 5.5 show the usability score given by users after using O-CAPTCHA. The CAPTCHA was rated on a scale of 1 to 5 where '1' is the easiest and '5' is the most difficult. The average score given of O-CAPTCHA after using it is 2.62 out of 5.

Figure 5.5 Statistical Overview



6. ROBUSTNESS TESTING FOR O-CAPTCHA

Robustness is the measure of how much the CAPTCHA is resilient against the automatic bot attacks. The O-CAPTCHA consists of text in black color and plain white background and CFS attack requires these conditions to work. Secondly Megaupload's dataset is used for testing, which can be broken by CFS Attack, so to test if the new layer gives O-CAPTCHA scheme strength O-CAPTCHA is passed through CFS attack.

6.1 CFS Attack on O-CAPTCHA

The reason for launching this attack on O-CAPTCHA is that it has been tested on Megaupload dataset in the underneath layer which can be cracked using this attack. Therefore CFA attack will show that if the orientation layer has made O-CAPTCHA resilient against segmentation attack like CFS. The CFS attack on the purposed CAPTCHA shows promising results, this CAPTCHA breaking tool fails in most scenarios presented by the proposed CAPTCHA model. Discovering and eliminating the conditions causing the CAPTCHA to fail makes the technique more robust. Few scenarios of this robustness test experiment have been discussed below.

6.2 Experimental Setupx for CFS Attack

For the CFS attack experiment a standard personal computer is required, no special hardware is required for this experiment. The attack has been programmed in C sharp language. There is no other specific software requirement. The programmed code takes the CAPTCHA image as input and shows segmented characters as output. The CFS segments one character from the CAPTCHA and gives it a color and next segmented characters in given some other color. Output shows multicolored characters; therefore the number of colors in the output image refers to the number of characters segmented by the algorithm.

6.2.1 CFS Attack Experiment: Case 1

In this case the input was "TSK1" broken down into six sub-images. 80% character "T" is in the first sub-image remaining part of "T" and 40% of the letter "S" is in the next sub-image. Third sub-image contains remaining part

of"S" and about 10% of the letter "K". Fourth sub-image contains more part of the letter "K". Fifth image contains the over lapping part of "K" about 10% percent of the Numeric Digit "1". Sixth and the last sub-image contain the remaining portion of the numeric digit "1". In this case the input contains broken letters and numbers and none of the characters in the challenge fully reside in one sub-image. It is shown in figure 6.1

Figure 6.1 Case 1

The output of the CFS shows that it has segmented ten characters but there are only four characters in this experiment. CFS failed to recognize the broken characters as one character and has recognized it as multiple characters.

6.2.2 CFS Attack Experiment: Case 2

In this case the input CAPTCHA containing four characters "FNW5" is divided into five sub-images. First sub-image contains the complete character's and about 20% of the character "N". Second sub-image contains the remaining portion of character "N" and about 20% of the character "W". Third sub-image contains most of the character "W". Fourth sub-image contains rest of the character "W" and some portion of the numeric digit "5". Fifth and the Last sub-image contains the remaining portion of the numeric character "5". Case 2 is shown in figure 6.2.

Figure 6.2 Case 2

The output of CFS attack shows that it has segmented nine characters instead of four which were given as input. Although CFS was able to correctly segment the last character the numeric "5" because it was in a separate sub-image but it fails overall because rest of the CAPTCHA was not correctly segmented.

6.2.3 CFS Attack Experiment: Case 3

In this case the input CAPTCHA contains the Letters "NXM7". It has been broken in to four sub-images. First sub-image contains 90% of the character "N" and second sub-image contains remaining part of character "N" and overlapping part of characters "N" and "X" along with rest of the character "X" and some initial portion character "M". Third sub-image contains most part of character "M". Fourth and the last sub-image contain remaining part of the Character "M" and the character "7". It is shown in figure 6.3.

Figure 6.3 Case 3

The output shows CFS has recognized six characters in the given CAPTCHA but the input contained not more than four characters. CFS has failed due to the reason or random distribution of portion of letters in different sub-images.

6.2.4 CFS Attack Experiment: Case 4

In this case the CAPTCHA containing the characters "CRP9" was divided into three sub- images. First sub-image contains entire character "C" and first line of the character "R". Second sub-image contains remaining part of alphabetic character "R" and some portion of the alphabet "P". Third and the last sub-image contains the remaining portion of alphabet "P" and the numeric character "9". It is shown in figure 6.4.

Figure 6.4 Case 4

The output of the CFS on this CAPTCHA challenge shows that it has segmented seven characters rather than four presented to it. The character 'P' is divided in a way that it looks like 'F' in second sub-image unless it is not properly oriented this 'P' will look like 'F' and mislead the bot. Humans on the other hand have the edge to use their knowledge of character orientation and put it into the right position and read the correct text. If this segmented character is given to any recognizer it will recognize it as the character 'F'. The character 'R' is also beyond recognition due to its division into two sub-images. The CFS however was able to segment two characters correctly.

6.2.5 CFS Attack Experiment: Case 5

In this case the input was DXD5 broken into two sub-images. This division was right in the middle of the two characters in a way that no character was broken into two pieces. Output shows that CFS has successfully segmented all four characters and broken this CAPTCHA challenge. There were four characters in CAPTCHA and CFS output image shows four colors, each character in different color. It is shown in figure 6.5.

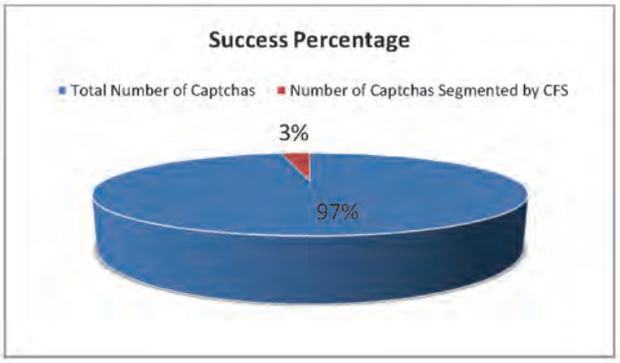
Figure 6.5 Case 5

The reason for the CFS success in this case is because the images have been divided in a way that it has not been able to break any character in-to two. This experiment proves that CAPTCHA is at risk of being cracked by the automatic Bot attack if the number of sub-images is only two.

6.3 Results of CFS Attack Experiments

Simple CFS algorithm has failed to crack the O-CAPTCHA. However in one case the CFS Attack proved to be successful. Results show insights in to scenarios that occur in the CFS experiments.

Figure 6.5 Success Percentage



The graph in figure 6.5 shows the overall segmentation success rate of the CFS attack. Out of one hundred CAPTCHA samples the CFS attack only able to segment three CAPTCHAs. With only 3% success rate the CFS has failed to segment any significant number of the new orientation CAPTCHAs. These three samples were the ones in which the number of sub-images were two as discussed in case 5. Let's look into the case 5 scenario.

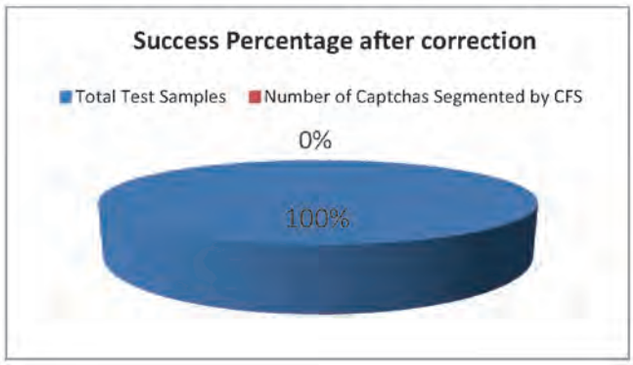
6.4 CASE 5 PROBLEM

This case occurs when the CAPTCHA containing four characters is divided into two sub-images. If the division of image does not divide any character into two, it is venerable to be segmented by the CFS Attack. Hence in this case, O-CAPTCHA is vulnerable to text based segmentation attack of CFS.

6.5 SOLUTION

To overcome this problem, we can redefine the range of number of sub-images in a challenge. The usability test gave results that 2 to 6 sub-images are an ideal range. Now modify this range and make it 3 to 6 and get maximum usability and robustness. In this way, the risk of being cracked by a bot become very close to zero. The graph in figure 6.6 shows that simply by changing the range of sub- images CFS attack fails to achieve any success.

Figure 6.6 Success Percentage after Correction



6.6 More Insights from CFS Attack Experiments

Using the Mega-upload dataset which contains 4 characters in every CAPTCHA it was observed in the experiment that odd number of sub-images perform better with even number of characters in sub-images. For example, the case with two sub-images and four sub-images more characters are in the condition as compared with the cases in with three and five sub-images.

6.7 More Attacks on O-CAPTCHA

Other attacks discussed in (Yan, J., 2008; Huang S., 2010; Ahmad A. el., 2012; Bursztein, E., 2011) are scheme dependent. It has been seen in the CAPTCHA breaking Experiments that those attacks failed on Meaga upload datasets. In proposed O-CAPTCHA we are using Megaupload's Dataset, so we can say that those attacks will fail on this modified version of Mega upload CAPTCHAs. Reasons of their failures are discussed below.

- Noise Extraction on PHP scheme-2 discussed in (Ahmad A. el., 2012) will fail because it targets the noise in CAPTCHAs and O-CAPTCHA's do not have noise.
- Background Extraction discussed in (Ahmad A. el., 2012) for Bot Detect CAPTCHAs will fail because O-CAPTCHA does not have any chess board background.
- K-mean clustering attack discussed (Li S., 2010) for breaking E-banking CAPTCHAs will fail on OCAPTCHA because O-CAPTCHA scheme does not have layers of text and background.
- OCR attack discussed in (Li S., 2010) for breaking E-banking CAPTCHAS fails as characters are broken and wrongly segmented OCR will fail to recognize.
- CFS attack of PHP scheme-1 discussed in (Ahmad A. el., 2012) will fail because O-CAPTCHA does not have noise like TV static.
- Dictionary attacks discussed in (Bursztein E., 2011) will not work on this scheme as O-CAPTCHA scheme do not use proper English words.

These attacks target the straight text or remove noise from the CAPTCHA. All of the attacks discussed earlier are incapable of putting the disorientated text in to proper shape. A bot intended to break O-CAPTCHA needs to first devise a method to put these broken characters to it proper position only then CFS could be successful to break them.

7. CONCLUSION AND FUTURE WORK

7.1 Conclusion

Giving the users a better experience with the interactive CAPTCHA gives the orientation CAPTCHA an edge over the other CAPTCHAs. The new orientation CAPTCHA has proved to be highly usable due to their controllable usability using the human intelligence. The unpredictability of the number of sub-images and the changing angle of rotation makes the new orientation CAPTCHA robust against the brute force attack. The broken characters feature makes it difficult for segmentation attacks i.e. CFS to segment the characters. The reason of CFS failure is that sub-images not always contain one full character some part of character in the next sub-image and CFS will recognize them as two separate characters. In addition to these security features that CAPTCHA inherits the original robustness of the text based CAPTCHA. In our study we have increased the robustness of the text based CAPTCHA by making them interactive without compromising on usability as the usability remains more or less same. Our techniques resists segmentation attacks because the characters are often broken and segmenting programs further segments that broken character in-to two characters and any recognition attack would consider it a noise, resulting in a robust CAPTCHA.

7.2 Future Work

The tug of war, between CAPTCHA breaking and secure CAPTCHA creation is going on for more than a decade. Now new usable and robust design comes on board and CAPTCHA breakers design bots to break them. This proposed orientation CAPTCHA promises to give the bot designers tough time because recent techniques to crack the CAPTCHA challenge fail on this CAPTCHA model. However in future work more intelligent bots can be designed to crack the proposed orientation CAPTCHA, after that O-CAPTCHA can be modified to be more secure against those automatic bot attacks. More future work can be carried out on the following lines of Action.

- Increase in number of Sub-images
- Changing underneath CAPTCHA
- Use of pictures in place of text CAPTCHA
- Merging O-CAPTTCHA with other CAPTCHA techniques

7.2.1 Increase in Number of Sub-Images

We can modify the scheme by increasing maximum the number of sub-images from 6 up to 12 with same number of scroll bars. Same principle of synchronized rotation of two sub-images can be scaled to synchronized rotation of

three or four sub-images. This is going to double the work for automatic bot attack. While the effort for the humans remain the same as the number of scroll bars they have to adjust are still three.

7.2.2 Changing Underneath CAPTCHA

In this research, Megaupload CAPTCHA is used as underneath. These CAPTCHAs have been chosen as their text is easy to read. However in future other text based CAPTCHA schemes can be used which are more usable for humans and more robust against automatic bot attacks. Combination of different CAPTCHA schemes can be used as underneath CAPTCHA so that every time user will be given different CAPTCHA to orient and read. This will make O-CAPTCHA even more unpredictable.

7.2.3 Use of Pictures Instead of Text

In this research, the underneath CAPTCHA is a text based CAPTCHA.In future, images can also be used using the same principle. The broken image will be shown to users and they put it together using scrollbars. This will be an interesting experiment as it will change entire shape of existing O-CAPTCHA scheme. The type of O-CAPTCHA will become a selection based picture CAPTCHA.

7.2.4 Merging with Other CAPTCHA Techniques

The concept of OCAPTCHA can be used with successful schemes like reCAPTCHA by giving user two O-CAPTCHA challenges to solve and these two words could be picked from old books need to be digitized.

Similarly the non-English CAPTCHA schemes can also be used as the underneath CAPTCHA for the added usability for segments of population who do not understand English language.

7.3 Concluding Remarks

Orientation based textual CAPTCHA is an entirely new domain of CAPTCHAs. This scheme is very scalable as there is a great amount of potential for further experiments. O-CAPTCHA has shown a very good performance in usability experiments in both smartphones and computer screen. So O-CAPTCHA can be used to protect valuable online resources.

REFERENCES

[1] Bairda, H., & Bentleyb, J. (2005). *Implicit CAPTCHAs*. In the Proceedings of SPIE/IS&T Conference on Document Recognition and Retrieval XII (pp. 191-196).

[2] Bursztein, E., & Bethard, S. (2009). *De CAPTCHA: Breaking 75% of Ebay Audio CAPTCHAs*. In Proceedings of the 3rd USENIX Conference on Offensive Technologies (pp. 8-18).

[3] Bursztein, E., Martin, M., & Mitchell, J. (2011). *Text-based CAPTCHA Strengths and Weaknesses*. In Proceedings of the 18th ACM Conference on Computer and Communications Security, (pp. 125-138).

[4] Chellapilla, K., Larson, K., Simard, P., & Czerwinski, M. (2005). *Computers Beat Humans at Single Character Recognition in Reading based Human Interaction Proofs (Hips)*. In Proceedings of the 2nd Conference on Email and Anti-Spam, (pp. 21-22).

[5] Chow, R., Golle., P., Jakobsson, M., Wang, L., & Wang, X. (2008). *Making CAPTCHA Clickable*. In Proceedings of the 9th Workshop on Mobile Computing Systems and Applications, (pp. 91-94).

[6] Desai, A., & Patadia, P. (2009). *Drag and Drop: A Better Approach to CAPTCHA*. In Proceedings of (INDICON), (pp. 1-4).

[7] El Ahmad, A., Yan, J., & Marshall, L. (2010). *The Robustness of a New CAPTCHA*. In Proceedings of the 3rd European Workshop on System Security, (pp. 36-41).

[8] El Ahmad, A. S., Yan, J., & Ng, W. Y. (2012). *CAPTCHA Design: Color, Usability, and Security*. In Proceedings of IEEE Internet Computing, 16(2), 44-51.

[9] Elson, J., Douceur, J., Howell, J., & Saul, J. (2007). *Asirra: A Captcha that Exploits Interest- Aligned Manual Image Categorization*. In Proceedings of the 15th ACM Conference on Computer and Communications Security (ccs '08). (pp. 535-542).

[10] Faymonville, P., Wang, K., Miller, J., & Belongie, S. (2009). *CAPTCHA-based Image Labeling on the Soylent Grid*. In Proceedings of the ACM SIGKDD Workshop on Human Computation, (pp. 46-49).

[11] Gossweiler, R., Kamvar, M., & Baluja, S. (2009). *What's up CAPTCHA? A CAPTCHA Based on Image Orientation*. In Proceedings of the 18th International Conference on World Wide Web, (pp. 841-850).

[12] Huang, S., Lee, Y., Bell, G., & Ou, Z. (2010). *An Efficient Segmentation Algorithm for Captchas with Line Cluttering and Character Warping*. In Proceedings of the Multimedia Tools and Applications, 48(2), 267-289.

[13] Kluever, K., & Zanibbi, R. (2008). *Video CAPTCHA: Usability vs. Security*. In the Proceedings of IEEE Western New York Image Processing Workshop.

[14] Li, S., Shah, S., Khan, M., Khayam, S., Sadeghi, A., & Schmitz, R. (2010). *Breaking E-banking CAPTCHAS*. in Proceedings of the 26th Annual Computer Security Applications Conference, (pp. 171-180).

[15] Matthews, P., & Zou, C. (2010). *Scene Tagging: Image-based CAPTCHA using Image Composition and Object Relationships*. In Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, (pp. 345-350).

[16] Baird, H. S., & Popat, K. (2002). *Human Interactive Proofs and Document Image Analysis*. Document Analysis Systems, (pp 3-4).

[17] Yalamanchili, S. & Rao, M. K. (2011). *A Framework for Devanagari Script-based CAPTCHA*. In Proceedings of the 19th ACM Conference on Computer and Communications Security, (pp.543-554).

[18] Nazir, M., Javed, Y., Khan, M., Khayam, S., & Li, S. (2011). *Poster: Captchœcker- Automating Usability-Security Evaluation of Textual CAPTCHAS*. In Proceedings of 7th Symposium on Usable Privacy and Security (SOUPS2011).

[19] Schlaikjer, A. (2007). A Dual-Use Speech CAPTCHA: Aiding Visually Impaired Web Users While Providing Transcriptions of Audio Streams, Technical Report CMU-LTI-07-014, Carnegie Mellon University.

[20] Simard, P., Szeliski, R., Benaloh, J., Couvreur, J., & Calinov, I. (2003). *Using Character Recognition and Segmentation to Tell Computer from Humans*. In Proceedings of the 7th International Conference on Document Analysis and Recognition, (pp. 418-423).

[21] Yan, J., & El Ahmad, A. (2007). *Breaking Visual CAPTCHA with Naive Pattern Recognition Algorithms*. In Proceedings of the 23rd Annual Conference on Computer Security Applications Conference, (pp. 279-291).

[22] Yan, J., & El Ahmad, A. (2008). *A Low-Cost Attack on a Microsoft CAPTCHA*. In Proceedings of the 15th ACM Conference on Computer and Communications Security, (pp.543-554).

[23] Ahn, L. V., Blum, M., Hopper, N. J., & Langford, J. (2003). *CAPTCHA: Telling Humans and Computers Apart*. In Advances in Cryptology, Eurocrypt, (pp. 294-311).

[24] Von Ahn, L., Maurer, B., McMillen, C., Abraham, D., & Blum, M. (2008). Re-CAPTCHA: Human-based character recognition via web security measures. *Science*, 321(5895), 1465-1468.

[25] Soni, R., & Tiwari, D. (2010). Improved CAPTCHA method. *International Journal of Computer Applications*, 1(25), 107-109.

[26] Shirali-Shahreza, M. H., & Shirali-Shahreza, M. (2006). Persian/Arabic CAPTCHA. *Journal of Universal Computer Science*, December, 12(12), 1783-1796.