DPCA: Dual Phase Cloud Infrastructure Authentication

R. Thandeeswaran[1], M A Saleem Durai[2]

[1]School of Information Technology and Engineering, VIT University, Vellore, India.
[2]School of Computing Science and Engineering, VIT University, Vellore, India.

Abstract: Every user appreciates the security of their data irrespective of its sensitivity. At the same time, user does not want to be held up with the legacy systems, which may be strongly secured but not with fantasy. Network travelled many phases of its own such as internet, intranet, wireless network, sensor network, Ad-hoc network, Mobile network, Blue-tooth, Cloud and IoT, the most popular technique now. Hence, whatever the type of network hitting the end user with improvised quality, get compromised in security.

People feel pleasure upon sharing their moments with their connections. Huge amount of data are in transit either by means of storing them in cloud or retrieving. Users lose control over their data, they are unaware about the people accessing and modifying. Hence DataCentres have to be protected from unauthorized illegal access. Illegal data access may be initiated by a boot or a bot. Man or a Machine discrimination has been resolved with CAPTCHA.

This paper, DPCA, proposes an authentication at two different phases. In the first phase, the user is authenticated with the new type of CAPTCHA. With this methodology, user is segregated as man or machine. Thereby bot-nets are filtered out and the flooding messages from bots are mitigated. In the second phase, where only man is allowed to access the cloud resources, he is authenticated with Dempster Shafter hypothetical approach combined with the user intent and not the content. DPCA has been tested in the infrastructure and the experimental results proved the strength of the algorithm.

Keywords: Dempster Shafer approach, Authentication, Security, CAPTCHA.

1. Introduction

Cloud infrastructure has excellent user-friendly applications with challenging performance. But when we discuss about the data storage, transparency is maintained. Users are unaware of where the data is stored, who is controlling and accessing the data, which leads to many undetected cyberattacks and social crimes. End-users are kicking off their data to cloud to reduce the burden of administering and to shed down the storage space crunch. All the data, sensitive or insensitive, are managed by the Cloud Service Providers (CSP). Any user can access the data and can either use or misuse. Hence, user access has to be controlled and unauthorized access to the datacenter is suppressed.

C-I-A (Confidentiality – Integrity – Access Control) is the basic requirement and fundamental demand for any application of data transmission to be secure.

Confidentiality is not revealing the user content to the third party, other than the communicating ends. Here the user verification provides a solution partly, which is obtained by proper user authentication.

Integrity is not modifying the data, time, and sequence in transit. Upon verifying the user, improper content modification can be limited. Again, user and message authentication could counteract against the attack on integrity.

Access control, inappropriate data retrieval and usage, can be achieved with the security service, Authentication. By properly authenticating the user, legitimate and illegal users can be identified and segregated. In this paper, at the initial stage of authentication, botnets are discriminated from end users. This user discrimination also aids in reducing the network traffic thereby congestion can be reduced. Hence, security is achieved without compromising the quality of service.

Authentication can be served at three different levels, such as User, message and machine.

- By validating user through user authentication, the man can be distinguished as Normal user/ legitimate user, Aggressive legitimate user / Flash crowd and an attacker. There are multiple user authentication methodologies, as shown in the comparative study. User authentication also enables the user and cloud service provider (CSP) to discriminate man and the machine. i.e. boot and bot.
- By validating the incoming message through message authentication, the falsification/modification of data can be verified. Discrepancy in the message authentication codes could help the receiver end to notify the error. Thereby message integrity can be achieved. Message encryption, Hash and MAC are the most prominent message authentication algorithms.
- By validating the machine through the common pair of addresses (MAC, IP), access through illegal/invalid addresses can be identified and blocked for imminent access.

The proposed work is structured as, Section 2 presents related work. Section 3 introduces the architecture overview and security issues of the newly proposed approach. Section 4 reveals the results observed during the simulation. Section 5 analyses the advantages of our approach and Section 6 concludes the work.

2. Related Work

Huge volume of data storage in DataCentres are not a concern. But securing them, from attackers (Authentication), illegal access (Access Control), modifying the content without the owner's knowledge (Integrity) and revealing the accessed data to the business competitors (Confidentiality) is major challenge in a cloud environment. Cloud Service Providers (CSP) try to provide a solution, but due to the universal fact that hacker is better than creator, the intruder could expose to the vulnerabilities. From end user work area to cloud, memory capacity was the limitation in the earlier

whereas data retrieval by the valid user is a major challenge. But in both the scenarios, protecting the data from illegal use and unethical access is dispute. Access to a sensitive data like bank account details, medical records and defense data must be restricted. Mere access privileges cannot serve the purpose. Identify the user before allowing him/her to access the data. Authenticating the user and authorize him to access the data every time could control the illegitimate accesses [1].

Authentication is the procedure to identify the right personality to access the data [3]. There are many factors that influence to verify the identity of the originator. Authentication takes multiple faces, as shown in fig.1, in the form of

- Password Authentication
- Biometric Authentication
- Multi-factor Authentication
- Phase Authentication
- Identity based authentication
- Recognizing and solving puzzle based authentication

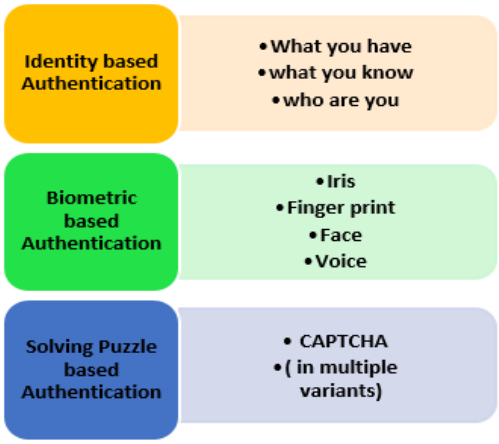

Figure 1. Multiple faces of Authentication

CAPTCHA – Complete Automated Public Turing Test to tell Computers and Humans Apart, is an example of challenge-response test used in computing to identify if the user is human. In 2000, Luis von Ahn, Manuel Blum, Nicholas J. Hopper of Carnegie Mellon University and John Langford of IBM came up with this term. The most common CAPTCHA requires users to type in the letters that were displayed. These letters had sometimes images and at times the letters or numbers jumbled. This type of CAPTCHA was the basis of all the currently present CAPTCHA and was invented by Mark D. Lilli Bridge, Martin Abadi, Krishna Bharat and Andrei Z. Broder [4].

The main use of CAPTCHAs is to prevent bots from using various computing services or collecting sensitive information like poll results, nominations and registering free email accounts or collect email addresses, and help prevent spam [5].Over the years, various versions of the CAPTCHA systems have been seen evolving from the basic idea, but all of them have lacked somewhere in terms of providing security.

Next, in 2007 reCAPTCHA, originated by Luis Von Ahn. It had the main aim of preventing spam and in the process of digitizing books. This worked on a simple principle of using humans to identify two words, one was distorted but could be read by a machine and the other scanned directly from books by an Optical Character Recognition (OCR) program. The concept was that if a human was able to complete the challenge of identifying the first word, the second word would be identified correctly. This second word would be presented to multiple users, the answer be compared and submitted to complete the digitization of the book. This improved security and also found a purpose of using this system. Till date, this is one of the most widely used CATCHA systems worldwide [21].

Table 1. Multi-factor authentication and their downsides

Method	Working Principle	Downsides
Multi-factor Authentication scheme for Cloud [6]	Works on the basis of dynamic secure multi-factor out-of-band secret-splitting mechanism	Assumes that all the new registrants are truthful, which is unimaginable.
Privacy preserving multi-factor authentication [7]	Assumed that intruder knows the victim's password and profile	Only brute-force and intent based attacks are focused.
Multi-level Authentication [8]	Generate one password and concatenate at several locations	Only Privacy is addressed.

After the development of reCAPTCHA, multiple complex systems evolved from the basic ideology and next came the Image CAPTCHA system. There are multiple variants of image based system. By types, it can be inferred that the type and the number of challenges to be solved was varying but the base principle of using images to identify humans persisted to be constant.

SILTCHA, used in the first phase of DPCA, is a combination of multiple such systems to provide a sustainable yet effective solution to bots and spam. It is then combined with Dempster's hypothetical approach in the second phase of DPCA to authenticate the users.

List of existing methodologies and their downsides are quoted in table1.

Most widely adopted authentication mechanisms are based on one of the following

- ***what you know*** includes :
  Passwords, PINs, Passphrases
- ***what you have*** includes :
  Smart cards, Magnetic cards, Hardware tokens, Software tokens
- ***who are you*** includes :
  Biometric components such as Palm prints, Fingerprints, Retina patterns and Hand geometry

DPCA solved the major challenges in authenticating users without compromising much of quality of service, the network traffic and congestion.

3. DPCA – Dual Phase Cloud infrastructure Authentication – Proposed approach

Dual Phase Cloud infrastructure Authentication has two phases, for verifying boot and for authenticating the user. The outline of the proposed approach is shown in figure 2.

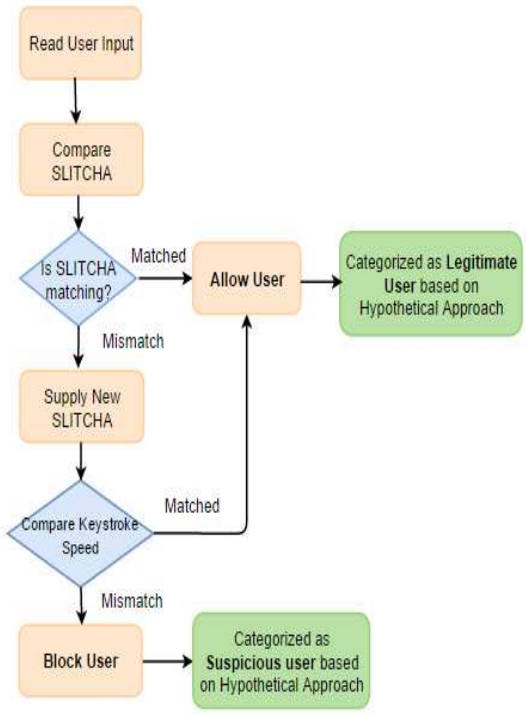

Figure 2. Overall work flow diagram of DPCA

3.1. Phase I: Discriminating Boot and the Bot using SILTCHA

Sentence, Image and Logic based Turing test to tell Computers and Humans Apart (SILTCHA), architecture and working offers a more intelligent way to access to the websites while preventing spam simultaneously. SILTCHA uses a sentence and image, co-related to each other, with the help of which the user needs to fill in the blank. Fig. 3 shows the SILTCHA as seen by any end user surfing over the browser.

Figure 3. Sample of SILTCHA as seen by user

SLITCHA proposed architecture is depicted in Fig. 4. When an end user requests for a web page the following eight step process takes place as well as shown in the figure

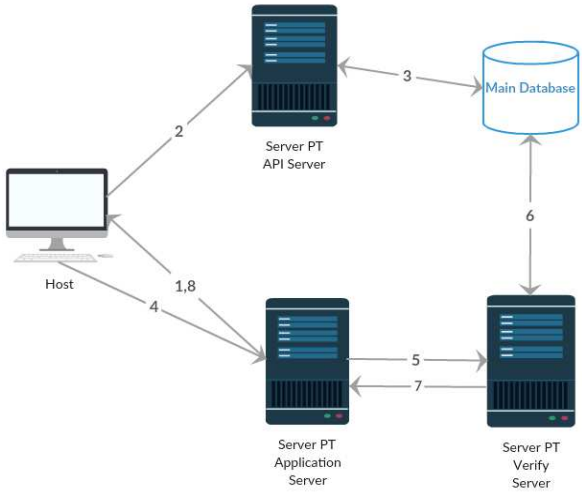

Figure 4. Architecture of SILTCHA

Using SLITCHA, the first phase discrimination of man and machine is performed. When the end user is a botnet, it will not wait for the response from the server. The Cloud Service Provider would treat each request as new and reserve resources for them, which ultimately deplete the server-side resources. CSP is unable to identify the request as only intentional traffic and not service-oriented requests. By filtering out the botnets, traffic rate can be reduced and also DC availability is improved.

Sequential steps followed in the proposal is prearranged below.

SILTCHA is further enhanced with keystroke technique. Pavithra, M. and KB Sri Sathya [9] presented an approach based on the keystroke. Keystroke technique is also a way of authentication. Typing speed of a user could reveal the genuine behavior of the user. Three steps have been followed here, such as

- User Enrolment: User is expected to fill various information about themselves. Objective is to observe the user's typing speed, the time duration

between two words, average typing speed for imminent words and the time duration of releasing keypad after typing words.

- User Verification: User verification performed based on the previously stored data. Data Capture, feature extraction and comparison of data from the stored data.
- User Identification: This phase allow users as an authenticated user only if the result matches with the stored result. If there is any discrepancy, user is block listed.

[1] **Host → Server**: the user requests a webpage.
[2] **User browser → API Server**: the user's browser requests a challenge SILTCHA from the API Server.
[3] **API Server → User browser**: API server retrieves a SILTCHA from the main database and relays it.
[4] User fills in the SILTCHA and submits the result to the Application Server.
[5] Application server then forwards this as a query to the Verify Server.
[6] Verify server does the verification based on the answer submitted by the user with the main database.
[7] Verify Server then sends the response to the Application Server.
[8] If the answer entered by the user is correct, it allows the user to proceed to the requested Web page otherwise a new empty SILTCHA is sent to the user for another attempt.

Bayesian principle, using probabilistic learning, continual incremental probability and standards, has been used for the feature extraction. MCMC (Markov chain Monte Carlo algorithm) used for the justification of authenticated user. It is basically used for generation of possible outcomes from an event. Like X, a random variable whose density (d) and distribution are already known. Hence, expected value generated for this variable is: $E[g(X)] = \int g(x)\, d(x)$.

3.2 Phase II: User Authentication

In the first phase with the help of SILTCHA with keystroke technique used for detecting whether a host is a boot or bot. This stage identifies any botnet based attempts to login to the system.

Second phase is user authentication, where the system finally verifies whether the boot is a legitimate or not. In this stage, a probabilistic approach has been used in learning about the user, based on his keystroke parameters which are fed to the Dempster–Shafer Algorithm (DSA). The output of the algorithm will tell us about the belief-plausibility interval related to that particular user. Based on this, legitimacy of the user can be predicted.

The SILTCHA reads a SUCCESS, when a user inputs the characters appearing on the screen within a fixed amount of time. If the input fails to read the characters in that time, a new SILTCHA appears with stronger security.

3.2.1 Hypothetical approach for user authentication

Based on Dempster Shafer approach, users are celebrated as legitimate or malicious with the help of user credentials. Attractive feature of this approach is, future outcome can be predicted. It differentiates the user based on the difference between belief and plausibility values. Belief is known as minimum probability of occurrence of any event while plausibility defines the maximum probability of occurrence of that event. If the difference between belief and plausibility is crossed a particular threshold limit then that event is considered as a suspicious event [14].

Three parameters are considered for identifying the user as legitimate or fraud.

1. *Consecutive key typing time (ckt):* calculated based on the fact of how much time the user consumes for typing a consecutive key.
2. *Delay key time (dkt):* defines how much time user took for typing new letter after releasing previous letters.
3. *Halt key time (hkt):* depicts the time taken by user on a particular letter.

Mechanism for hypothetical approach of user authentication

Step1 On the basis of user accessing of information, assign the mass value (is known as the given probability for particular event) for ckt, dkt, hkt.

Step2 With the given events, generate the power set of events. Like,

$$p(ckt, dkt, hkt) = \{\emptyset, \{ckt\}, \{dkt\}, \{hkt\}, \{ckt, dkt\}, \{ckt, hkt\}, \{dkt, hkt\}, \{ckt, dkt, hkt\}\}$$

Step3 For the power set elements, generate the mass values with the help of this mathematical approach.

$$M(a, b) = m(a) + m(b) - m(a)m(b)$$

$$M(a, b, c) = m(a) + m(b) + m(c) - m(a)m(b) - m(b)m(c) - m(a)m(c)$$

Step4 Calculate the belief for all the elements of power sets.

Belief of an event is known as the sum of the masses of event which are the subsets of that particular event. From power set of elements,

$$Bel(ckt) = m(\emptyset) + m(ckt)$$

$$Bel(ckt, hkt) = m(\emptyset) + m(ckt) + m(hkt) + m(ckt, hkt)$$

Similarly belief value is computed for all the events.

Step5 Calculate the plausibility for all the elements of power sets.

Plausibility of an event is known as the sum of all masses of the sets that intersect with that particular event.

$$Pl(ckt) = m(ckt) + m(ckt, dkt) + m(ckt, hkt) + m(ckt, dkt, hkt)$$

$$Pl(ckt, dkt) = m(ckt) + m(dkt) + m(ckt, dkt) + m(dkt, hkt) + m$$

Plausibility value for all other events is calculated in the same manner.

Step6 Calculate the differences between belief and plausibility of the events.

Step7 Compare the differences between belief and plausibility with the threshold value.

Step8 If the difference is greater than threshold value then that events will be in suspicious phase otherwise event is in normal phase. All these possible events provide a degree of truth, based on which the suspicious event can be judged.

4. Experimental Results

An experimental study was done with the help of more than fifty end-users who tested the first time implementation of SILTCHA. The system was designed using PHP as the front end and MySQL as the backend to store and manage the database.

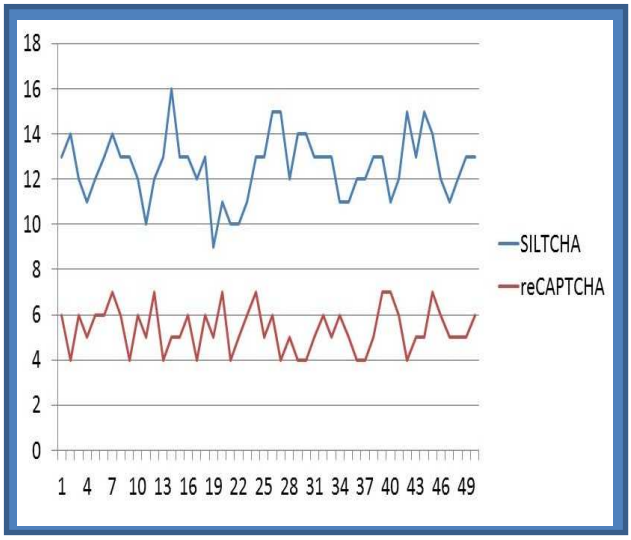

Figure 5. Comparison of SILTCHA and reCAPTCHA solved in one minute by 50 users

The database consisted of 100 sentences to image associations that were called from the database in the form of a query and presented to the user to solve. Once the user responded to the SILTCHA displayed, the response was then verified with the database to check if the user is human based on the correctness of the answer. To successfully pass the test, the user needs to identify the object in the image and fill in the blank with the appropriate word consisting of the exact number of alphabets or characters as shown in the sentence. This system was made available using a proxy server enabling users to access the local host of the main system containing the PHP – MySQL bundle.

The study consisted of 100 SILTCHA challenges that were created and presented to the 50 users. They were to solve as many as they could in one minute which averaged to 13 SILTCHAS per minute. The users out of 650 attempts solved 597 correctly (~92%). The failure rate was caused due to inability to associate the correct form of the word to match the image and also a very few cases where the text was unreadable by the user.

Table 2 presents the comparison among the various CAPCTCHA with SILTCHA systems available based on different parameters.

The experimental results carried out are summarized in the following three graphs. Fig. 4 is a representation of the total number of SILTCHA verses reCAPTCHA solved in one minute by 50 users. It can clearly be concluded that SILTCHA is a faster system as compared to reCAPTCHA.

Table 2. Comparison of various CAPTCHA systems and how SILTCHA fares against them

Parameter	Image Base	Math Base	Video	Re-CAPTCHA	SILTCHA
Security	Average	High	Average	High	**High**
Ease of Use	Easy	Hard	Hard	Hard	**Average**
Bandwidth Usage	~8KB	~5KB	~600KB	~5KB	**~12KB**
Time to Solve	Depend	~10 secs	~15 secs	~8 seconds	**~5 seconds**
Challenges to Solve	**Depend**	**1**	**1**	**2**	**1**

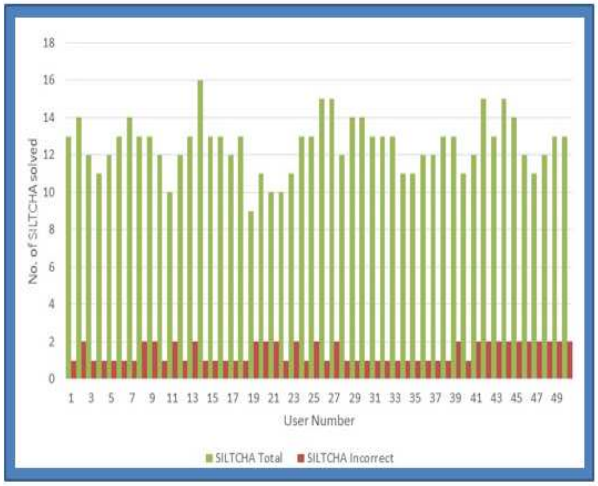

Figure 6. Comparison between the total solved SILTCHA and incorrectly solved SILTCHA

Fig. 6 is a representation on a graph of the total number of SILTCHA solved and the incorrectly solved SILTCHA. While comparing it with Fig.7 which shows the same values for reCAPTCHA, we see that a user can solve lesser reCAPTCHA with more percentage of failure as compared to SILTCHA.

The differences seen in Fig.6 and Fig.7 make it very evident that SILTCHA has a fair lead over reCAPTCHA in term of user solvability and overall correctness.

Figure 7. Comparison between the total solved reCAPTCHA and incorrectly solved reCAPTCHA

5. Conclusion

Secure user authentication has been a concern for many years. Many new solutions have come up with their own merits and demerits. One such solution makes use of

CAPTCHAs (Completely Automated Public Turing Test to Tell Computer and Humans Apart) to tell whether the host which is accessing the system is human/bot. In this paper, we have proposed a new framework for identifying against human/bots.

CAPTCHAs have been around for quite some time and will continue to evolve. There have been many innovative methods which have been incorporated into CAPTCHA systems. SILTCHA, our proposed system, has proved to be very effective, easy to use, and low on bandwidth usage which means that it is mobile data user friendly. According to the values and data in Section 4, it is a very practical and effective system. SILTCHA takes an average of five seconds to solve and it continues to be a fun way of preventing bots and spam while allowing human users to gain access to the web pages that they have requested.

All that said, we found that the database at the moment consists of only 100 entities that are randomly generated and called to the SILTCHA interface when requested by the user and this database will be ever evolving to grow to a larger collection of data. Since the system is developed in PHP and MySQL, we have experienced a little difficulty in migrating this system from one server to another but with a little knowledge in both the domains; we were able to overcome this and able to protect the resources from bot attacks.

A new SILTCHA combined with key stroke technique was proposed. Also, as a step ahead, we have also made use of a hypothetical approach, based on Dempster-Shafer Algorithm (DSA). The algorithm gives us a belief interval to predict whether the user who passed the SILTCHA test is a legitimate/malicious one based on his keystroke characteristics.

Due to the limitations in applying the DSA algorithm in practical, we had made use of only three features. Also, the SILTCHA's character set is limited and have some performance issues when implemented. In future, we would be extending the system to include more keystroke features, as well as, improve on the limitations faced.

References

[1] R. Thandeeswaran, M A Saleem Durai, Wide-ranging Survey on Authentication Mechanisms, International Journal of Applied Engineering Research, Vol.11, No.6, 2016, pp.4114-4117

[2] S. Khan, K. K. Loo, “Real time cross layer flood detection mechanism,” Elsevier Journal of Network Security, Vol. 16, No. 5, pp. 2-12, 2009.

[3] Michael E. Whitman, Herbert J. Mattord, Principles of network security, 4th edition, 2012.

[4] http://en.wikipedia.org/wiki/CAPT CHA

[5] M. Babaei, M.B. Ghaznavi Ghoushchi, A. Noori, “YAPPTCHA: Yet Another Picture Promoted CAPTCHA with Spam Stopping, Image Labelling and SIFT Accomplishment”, IEEE, 2013.

[6] Rohitash Kumar Banyal, Pragya Jain, Vijendra Kumar Jain, Multi-factor Authentication Framework for Cloud Computing, IEEE Fifth International Conference on Computational Intelligence, Modelling and Simulation, 2013, pp. 105 – 110.

[7] Wenyi Liu, A. Selcuk Uluagac, and Raheem Beyah, MACA: A Privacy-Preserving Multi-factor Cloud Authentication System Utilizing Big Data, 2014 IEEE INFOCOM Workshop on Security and Privacy in Big Data, pp. 518 – 523.

[8] Dinesha H A, Agrawal V K, Multi-level Authentication Technique for Accessing Cloud Services, 2012 IEEE, International Conference on Computing, Communication and Applications.

[9] M. Pavithra, KB Sri Sathya, Continuous User Authentication Using Keystroke Dynamics, International Journal of Computer Science and Information Technologies, Vol. 6, No.2, 2015, pp.1922-1925.

[10] N. Jeyanthi, Hena Shabeeb, R. Thandeeswaran, M Saleem Durai, RESCUE: Three Phase Authentication to Detect and Prevent DDoS attacks in Cloud Computing Environment, International Journal of Engineering, Transaction B: Applications, Vol.27, No.8, pp.1137-1146, 2014.

[11] Zhu, B., Captcha as Graphical Passwords–A New Security Primitive Based on Hard AI Problems, IEEE Transactions on Information Forensics and Security, Vol.9, No.6, 2014, pp. 891-904.

[12] N. Jeyanthi, Uttara Barde, M. Sravani, Venu Tiwari, N.Ch. S. N. Iyengar, “Detection of Distributed Denial of Service attacks in cloud computing by identifying spoofed IP”, International Journal of Communication Networks and Distributed Systems, Vol.11, No. 3, pp.262 – 279, 2013.

[13] Ragavi, V., and G. Geetha., CAPTCHA Celebrating its Quattuor decennial-A Complete Reference, International Journal of Computer Science Issues, Vol.8, No.6, 2011.

[14] Glenn Shafer: A Mathematical Theory of Evidence. ISBN: 9780691100425314

[15] Goswami, Gaurav, Face DCAPTCHA: Face detection based color image CAPTCHA, Future Generation Computer Systems Vol. 31, 2014, pp. 59-68.

[16] Truong, Huy D. and Christopher F. Turner, and Cliff Changchun Zou, iCAPTCHA: the next generation of CAPTCHA designed to defend against 3rd party human attacks, IEEE International Conference on Communications, 2011.

[17] Khan, Shafiullah, Khan Pathan, Wireless Sensor Networks: Current Status and Future Trends, Taylor and Francis (CRC) Publisher USA, 2012

[18] N. Jeyanthi, N.Ch.S.N.Iyengar, P.C.M.Kumar, A. Kannammal, “An Enhanced Entropy Approach to Detect and Prevent DDoS in Cloud Environment," International Journal of Communication Networks & Information Security(IJCNIS), Vol. 5, No. 2, pp. 110-119, 2013.

[19] N. Jeyanthi, N.Ch.S.N.Iyengar, Packet Resonance Strategy: A Spoof Attack Detection and Prevention Mechanism in Cloud Computing Environment, International Journal of Communication Networks & Information Security, Vol. 4, No. 3, pp. 163-173, 2012.

[20] Hicham Toumi, Amal Talea, Bouchra Marzak, Ahmed Eddaoui, Mohamed Talea, Cooperative Trust Framework for Cloud Computing Based on Mobile Agents, International Journal of Communication Networks & Information Security, Vol. 7, No. 2, pp. 106-115, 2015.

[21] http://www.google.com/reCAPTCHA/ intro/ index.html