International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue III, March 2018- Available at www.ijraset.com

Honeypot Captcha against Spam bots

Remya Vinayakumar[1], Alisha Nelson[2], Diana P.A.[3], Emilda Jojo[4], Ranjith R[5], Sheril Johnson[6]
[1]*Assistant Professor, University of Calicut, Computer Science Department, I E S College of Engineering, Chittilappilly, Thrissur, Kerala*
[2, 3, 4, 5, 6]*University of Calicut, Computer Science Department, I E S College of Engineering, Chittilappilly, Thrissur, Kerala*

Abstract: The Invisible Captcha control plays upon the fact that most comment spam bots don't evaluate JavaScript. However, there's another behavioural trait that bots have that can be exploited due to the bots inability to support another browser facility. When spam bots encounter a form field, they go into berserker frenzy try to fill out each field. At the same time, spam bots tend to ignore CSS. A honey pot is a field added to the form that the users can't see due to CSS or JavaScript (which hides the field). They don't inconvenience users like a captcha and they are a valid tool for thwarting spam bots. Basically, a spam bot fills in a field that valid users can't see, alerting us to their activity. If the honey pot field is filled in we can confidently reject the form as spam.

Keywords: Captcha, Berserker Frenzy, CSS, JavaScript, Spam Bots

I. INTRODUCTION

A spam bot is a computer program designed for sending spam. Spam bot usually create accounts and send spam messages with them. Web hosts and website operators have responded by spammers, leading to an ongoing struggle between them. Spammers find innovative ways to evade the bans and anti-spam programs, and hosts counteract these methods. We create a honey pot with the same name as one of the default and make it look legit with a label. Then we will place the honey pot in the form in a random location. It will keep moving it around between the valid field. So that the spam bot writer will not simply ignore the same field based on index. Now the valid fields now begin to look like honey pot to the spam bot we will add an expression to our form. This will keep spam bot from using the same fields and submitting the form later.

We must hide the honey pot to keep the valid users from filling it out. Honey pot can be done in java or CSS. When the bots make their entries, their data are being stored into a location. They will never able to try it again because data about them are already stored. A honey pot trap involves creating a form with an extra field that is hidden to human visitors but readable by robots.

The robot fills out the invisible field and submits the form, leaving you to simply ignore their spams submission or blacklist their IP. It's a very simple concept that can be implemented in a few minutes. Along with that an online voting system is been implemented with the security of honey pot captcha where voting can be done online using their Aadhar cards and convincing credentials. The candidates and voters can register online and voting will be completely conducted online. The results will be also published.

II. LITERATURE SURVEY

In the work called A ZigBee Honeypot to Assess Iot Cyberattack Behaviour, Seamus Dowling, Michael Schukat and Hugh Melvin introduce a novel method in which a honeypot that simulates a ZigBee gateway is created. It is designed to assess the presence of ZigBee attack intelligence on a secure shell attack vector. It captures all attack traffic for retrospective analysis. It sandboxes attacks of interest to determine if any attempts are targeting ZigBee specifically. Finally, it concludes that all captured mass attacks are mainstream distributed denial of service and bot malware, whereas individual attackers were attracted to and interacted with the ZigBee simulated Honeypot. Another combined work by group of researchers, Abigail Paradise, Asaf Shabtai, Rami Puzis, Aviad Elyashar, Yuval Elovici, Mehran Roshandel and Christoph Peylo is called Creation and Management of Social Network Honeypots for Detecting Targeted Cyber Attacks. In that work they propose a framework for management of social network honeypots to aid in detection of APTs (Advanced Persistent Threat) at the reconnaissance phase. In it they discuss the challenges that such a framework faces, describe its main components, and present a case study based on the results of a field trial conducted with the cooperation of a large European organization. In the case study, they analyse the deployment process of the social network honeypots and their maintenance in real social networks. The honeypot profiles were successfully assimilated into the organizational social network and received suspicious friend requests and mail messages that revealed basic indications of a potential forthcoming attack. In addition, they also explore the behaviour of employees in professional social networks, and their resilience and vulnerability toward social network infiltration.

In another paper named Use of Honeypots for Mitigating Dos Attacks Targeted on Iot Networks, researchers M. Anirudh, S Arul Thileeban and Daniel Jeswin Nallathambi introduces a honeypot model for mitigating DoS attacks launched on IoT devices. Honeypots are commonly used in online servers as a decoy to the main server so that the attack is mitigated to the decoy instead of the main server. Here a similar methodology is used to avoid the whole IoT system from being shut down due to a DoS attack.
Finally, Jana Medkova, Martin Husak, Martin Vizvary and Pavel Celeda in their work called Honeypot Testbed for Network Defence Strategy Evaluation describes a network defence strategy testbed, which could be utilized for testing the strategy decision logic against simulated attacks or real attackers. The testbed relies on a network of honeypots and the elevated level of logging and monitoring the honeypots provide. Its main advantage is that only the decision logic implementation is needed to test the strategy. The testbed also evaluates the tested network defence strategy. They demonstrate an example of network defence strategy implementation, the test setup, progress, and results.

III.PROPOSED SYSTEM

This section explains the proposed system along with Proposed Architecture, Login Form, Honeypot Trap and Online Voting System.
In proposed system, we Create a honey pot with the same name as one of the default and Make it look legit with a label. It is invisible to human eyes. Then we will Place the honey pot in the form in a random location and will Keep moving it around between the valid fields. A user first tries to login to the online voting portal by signing up their data. When any automated programmes or a spambot tries to fill these fields, they will be restricted from the site by showing the alert message. We can simply filter the real user from the spam using this hidden field. After signup the admin will verify the credentials by the user. The field officer is the one who approves each entry. admin will set a start voting after which the voters can vote and he can also stop the voting after time.at the end of voting the results will be notified at a preannounced date.

A. Proposed Architecture

The Figure 1 illustrates the architecture of our system. There are mainly two parts of the system.one is the login form part where honeypot is deployed.

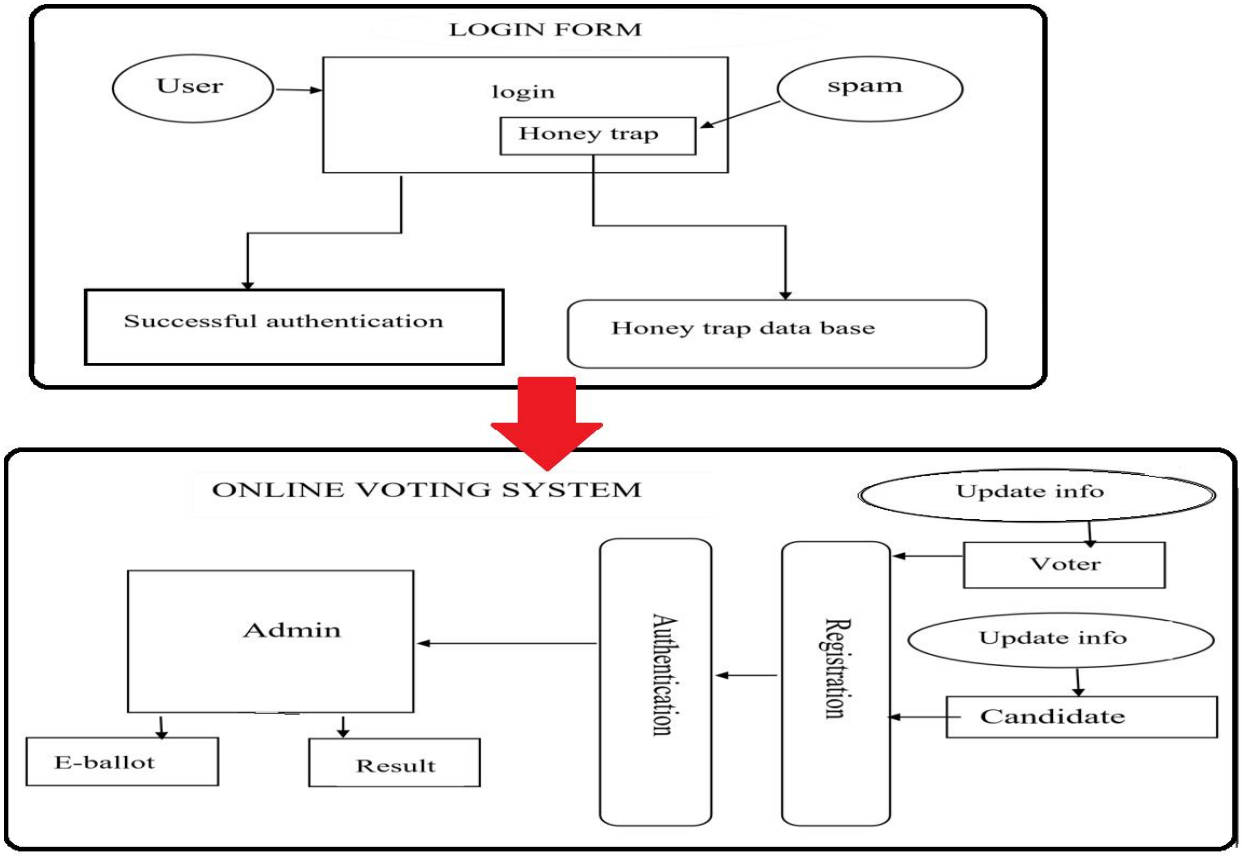

Fig.1 System Architecture

B. Login Form

HTML forms provide a simple and reliable user interface to collect data from the user and transmit the data to a server for processing. Server simply reads all the HTTP data sent to it by the browser, then returns a Web page. HTML forms helps to create variety of user interface controls to collect input in a Web page. Each of the controls typically has a name and a value, where the name is specified in the HTML and the value comes either from user input or from a default value in the HTML. The entire form is associated with the URL of a program that will process the data, and when the user submits the form (usually by pressing a button), the names and values of the controls are sent to the designated URL as a string of the form.

C. Honeypot Trap

A honey pot trap involves creating a form with an extra field that is hidden to human visitors but readable by robots. The robot fills out the invisible field and submits the form, leaving one to simply ignore their spams submission or blacklist their IP. It's a very simple concept that can be implemented in a few minutes and can add them to contact and submission forms to help reduce spam.

D. Online Voting System

The proposed Online Voting system using OTP with Aadhar Id and pseudorandom number generator. In this proposed method, we will commence the work with database creation. To create the database, we collect the voter detail from different voters. Have a user-friendly interface and user guides understandable by people of average computer skills. Be robust enough so that users do not corrupt it in the event of voting. Can handle multiple users at the same time and with the same efficiency, this will cater for the large population.

1) *Administrator:* The election commissioner or admin have the authorization to access this one. Admin is the super user of the system. admin or election officer will get the requests from the General users and Nomination candidates. In this module admin will receive the all registration request from users. These requests will be scrutinized by admin. Admin should give the acceptance to the candidates those who have send for party candidates register. Nomination acceptance will be finalized by the admin itself. Administrator can view the all users, Candidates by constitution wise in the form of reports.
2) *Registration:* Through this general user can registered himself as well as candidate. The user should provide the entire information about him regarding address, contact no, e-mail id etc. User should upload this image at the time of registration.
3) *Candidate:* To the registration as party candidate, that the candidate should be registered first. The party registration request goes to admin. The acceptance will be given by admin only. Candidate can enter his home page by using his credentials that was accepted by admin. In this module candidate would update his profile like his experiences, his promises if any. From this interface, he can make a nomination. Nomination will be accepted or reject by admin.
4) *Voter:* As a general user, he can visit the site. If the user wants to vote registration then he enters to registration interface from this can make a request to the system or admin providing all information about including his photo, address details. The request will be going to Admin. The User must wait up to verification. After Verification has done successfully then can have rights to access or enter his home page. By user interface he can make vote for required candidate on election Date.
5) *Authentication*: This contains all the information about the authenticated users. User without his username and password can't enter the login if he is only the authenticated user then he can enter to his login and then he will have authorization based upon their roles.

IV. IMPLEMENTATION

Web based application are often viewed as simply a form of software systems. To remove defects effectively, at the earliest possible stage and at lowest cost, software development, acquisition, or supply involve both verification and validation activities is on actual testing (unit, system, integration and acceptance testing) rather than review or inspections. Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users, which it will work efficiently and effectively. It involves careful planning, investigation of the current System and its constraints on implementation, design of methods to achieve the changeover, an evaluation, of change over methods. Apart from planning major task of preparing the implementation are education and training of users. The more complex system being implemented, the more involved will be the system analysis and the design effort required just for implementation.

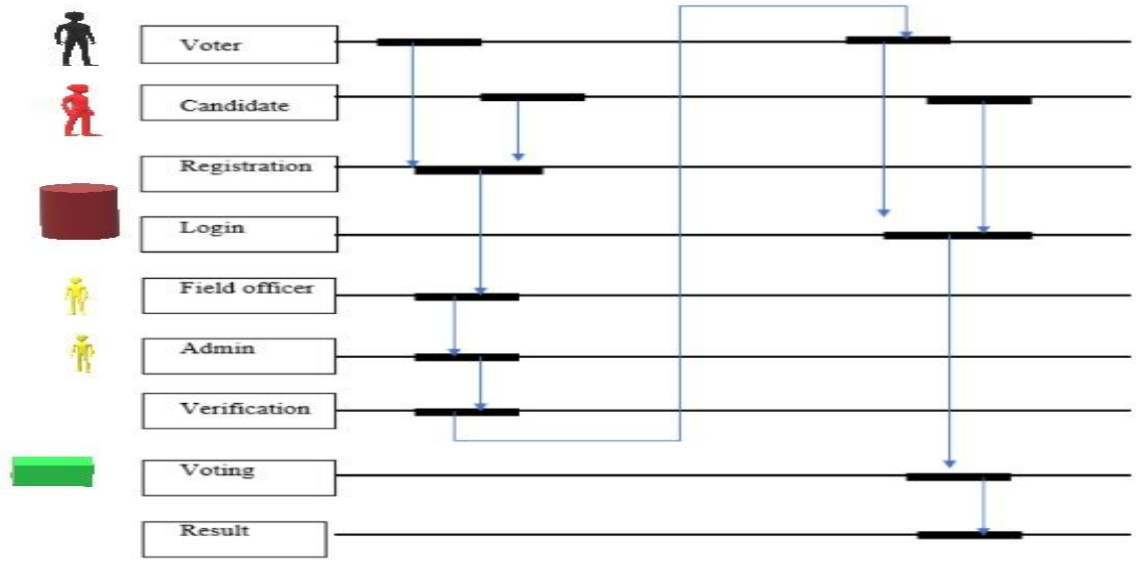

Fig.2 Modules sequence diagram

The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions made regarding the equipment and resources and the additional equipment must be acquired to implement the new system. Figure 2 describes the implementation through a module sequence diagram. Implementation is the final and important phase, the most critical stage in achieving a successful new system and in giving the users confidence. That the new system will work be effective. The system can be implemented only after through testing is done and if it found to working according to the specification. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle certain type of transactions while using the new system.

V. CONCLUSIONS

Honey pot trap observes the protection of network resources from attacks. As an application can use honey pot trap in captcha generations. This technique is used to trap the spammers. All the information about the spammers are caught in the honey pot. These data are used to block them from further attack. To show how it works we have attached an online voting application in to which we login. This will make us aware of how this entire system of honey pots works in case of security of data. Using honey pot captchas in future will reduce our effort in logging into web pages. It will save our time and more secure method security for captchas.

VI. ACKNOWLEDGMENT

We owe a debt of gratitude to Dr. S. Brilly Sangeetha-Head of Computer Science Department, for the vision and foresight which inspired us to conceive this project. We also show our sincere thanks to Ms. Remya Vinayakumar who has been guiding us in moulding this project into a successful one.
Above all, we are very much thankful to the Almighty God for showering his blessings upon us for this great success.

REFERENCES

[1] Aleksey A, Sergey V, Igor M, “Development and implementation of a honey trap” US Patent 978-1-5090-4865-6, Jan 2017.
[2] Anirudh M, “Use of Honeypots for Mitigating DoS Attacks targeted on IoT Networks” IEEE International Conference on Computer, Communication and Signal Processing (ICCCSP-2017).
[3] Atzori A. Iera G. Morabito "The internet of things: A survey", Computer networks vol. 54 no. 15 pp. 2787-2805 2010. "6.4 Billion Connected “Things" WillBeinUsein2016[online]Available: http://www.gartner.com/newsroom/id/3165317
[4] Atzori, L., Iera, A. and Morabito, G., 2010. “The internet of things: A survey”. Computer networks, 54(15), pp.2787-2805.
[5] Bellovin S.M. "Packets found on an internet" SIGCOMM Comput. Commun. Rev. vol. 23 no. 3 pp. 26-31 1993. "Iot devices found to carry out ddos attacks" Computer Security Update 1 Oct. 2016
[6] Jana Medková_y, Martin Husák_y, Martin Vizváry_, Pavel C˘ eleda_, “Honeypot Testbed for Network Defense Strategy Evaluation” US Patent 978-3-901882-89-0, Jan 2017 IFIP.
[7] Kouil T. Rebok T. Jirsík J. Cegan M. Drasar M. Vizváry J. Vykopal "Cloud-based testbed for simulation of cyber-attacks" Network Operations and Management symposium(NOMS) 2014 IEEEpp.1-6May2014.

[8] Medková M. Husák M. Drašar "Network defense strategy evaluation: Simulation vs. live network" Proceedings of the 2017 IFIP/IEEE International Symposium on Integrated Network Management IM 2017 2017.
[9] Provos "A virtual honeypot framework" SSYM'04: Proceedings of the 13th conference on USENIX Security Symposium.
[10] Seamus Dowling, Michael Schukat, "A ZigBee Honeypot to assess IoT Cyberattack Behaviour" IEEE Journal of Computer Engineering, vol.16, pp.73–80, Mar. 2017.
[11] Z. Feng, Y. Zhu, P. Xue, and M. Li, "Creation and Management of Social Network Honeypots for Detecting Targeted Cyber Attacks," in IEEE Transactions on Computational Social Systems 2017, Jul.
[12] Zhu R. Wang Q. Chen Y. Liu W. Qin "IOT gateway: Bridging wireless sensor networks into Internet of Things" ed. Proceedings of the IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing 2010. IEEE 802.15.4 Standards and