JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Honey-Pi: A Honeypot installed on Raspberry-Pi

Prabhakar Pal [1] , Vinit Mehta [2] , Pratik Nimbalkar [3] , Prathamesh Dodia [4] , Supriya Dicholkar [5]

[1]*BE Electronics and Telecommunication Engineering Student, Atharva College of Engineering, Mumbai, India, Mumbai University*

[2]*BE Electronics and Telecommunication Engineering Student, Atharva College of Engineering, Mumbai, India, Mumbai University*

[3]*BE Electronics and Telecommunication Engineering Student, Atharva College of Engineering, Mumbai, India, Mumbai University*

[4]*BE Electronics and Telecommunication Engineering Student, Atharva College of Engineering, Mumbai, India, Mumbai University*

[5]*BE Electronics and Telecommunication Engineering Assistant Professor, Atharva College of Engineering, Mumbai, India, Mumbai University*

Abstract:

In the modern times with an exponential increase in digitization, the world is reaching new heights in computerization and networking as each and every being is interconnected to each other via some network to increase efficiency and advancements. The digital era has many merits which supports the new technological demands of the world, but with merits there are also obstacles which might hinder the usual workflow of the environment. Many intruders, hackers or unethical users of this digital platform can exploit to harm the system or network of another person. These intruders generally hack the system for a ransom in return. In order to minimize this threat, it is necessary to have a security system that has the ability to detect attacks and block them. A honeypot is one of those counter-measures against the intruders which would defend our systems and also provide sufficient data about the attack. An IDS or IPS can also provide sufficient defense but it won't be efficient to provide data about the attack. Generally, Honeypots are deployed so that they can give out information about the attack like IP address, packets sent to the network, et cetera.Honeypots are deployed in various network environments such as military, commercial and now-adays also personal to expand the security measures for IOT devices.

Keywords: ***Honeypot, Intrusion Detection System, Intrusion Prevention System, Security, Intruder***

I. Introduction

A honeypot provides a defense mechanism against attackers or intruders who try to break into the system and might damage the system. It is a well designed system that attracts hackers into it. By luring the hacker into the system, it is possible to monitor the processes that are started and running on the system by the hacker. In other words, a honeypot is a trap machine which looks like a real system in order to attract the attacker. Honeypot is a great way to improve network security administrators' knowledge and learn how to get information from a victim system using forensic tools. Honeypot is also very useful for future threats to keep track of new technology attacks. We can divide honeypots according to their aims and level of interactions. If we look at the aims of the honeypots, we can see that there are two types of honeypots, which are research honeypots, and production honeypots:

i) Research Honeypots: Research honeypots are mostly used by military, research and government organizations. They are capturing a huge amount of information. Their aim is to discover new threats and learn more about new Blackbox approaches and techniques. The objective is to learn how to protect a system better, they do not bring any direct value to the security of an organization.

ii) Production Honeypots: Production honeypots are used to protect the company from attacks, they are implemented inside the production network to improve the overall security. They are capturing a limited amount of information, mostly low interaction honeypots are used. Thus, the security administrator watches the hacker's movements carefully and tries to lower the risks that may come from it towards the company.

As we categorized honeypots according to their aims, now we can look into more details in levels of interactions. Level of interaction stands for how much the hacker will be able to interact with the system. More amounts of data we would

like to gather require more levels of interaction. More level of interaction brings more risks into the network security as well. Based on the needs and the purpose of the experiment that one would like to examine, there are mainly two levels of interaction that are low level and high level.

i) Low-Level interaction: In low level honeypots one can get the least amount of data compared to other honeypot systems. They are limited, so the risk that was taken from intruders is not big either. Here, there isn't any operating system to deal with. They can be used to identify new worms or viruses and analyze the traffic that is going on through the network.

ii) High-Level interaction: High interaction honeypots are the most advanced honeypots. Unlike low interaction honeypots, there is an operating system. As a consequence, the hacker can perform anything. Proportionally, more data can be captured from the hacker's activities. These kinds of honeypots are very time consuming and difficult to deploy and maintain as well.

Now coming to why the name Honey-Pi, a honeypot which would be installed and configured on a Raspberry Pi. A Raspberry Pi would effectively reduce the cost, as setting up a proficient server will cost around INR 30K whereas a raspberry pi will reduce it by more than 50%. A honeypot can also be installed on the regular windows computer but that would cost high power consumption and those computers also take ample amount of space which might be an issue in a place where there is scarcity of space.

II. Related Work

The previous section gave us an idea about what a honeypot is and how it functions. This section will address more about the work done in this field and related research on decoys in different areas to combat the attackers. [1] Aparna Tiwari and Dinesh Kumar have compared various honeypots based upon their parameters. These parameters clearly explain the pros and the cons of these honeypots. Honeypots are classified on the basis of purpose and their level of interaction. On the basis of purpose, it is classified into production and research purpose. Production based honeypots are usually placed inside the production network along with other production servers by an organization to enhance their security. Example: Netbait, etc. research-based honeypot is used to gather information about the motives and tactics of the black hat targeting in different networks. Example: Bigeye, etc. Honeypots are also classified on the basis of their interaction level. There are three types of interaction level: low, medium and high. In low level the Interaction of black hats with the system is limited and for small time thus black hats cannot intrude the system. In medium level interaction it exists as a middle ground between low & high interaction solutions. High level interaction provides Maximum information of black hats allowing them to access the whole system or even tamper it. There are various tools deployed depending upon the level of interaction of the honeypot. For ex- KFsensor, Netbait, Mantrap, Honeynets. Paper has described a list of various honeypots in a comparative manner so that a user can easily understand which one is preferable for them. Many honeypots are still developing to meet the requirements of the organizations so that they can fully understand the type of attacker, their intention of hacking or attacking.

[2] Further, CR Hecker has deployed a dynamic honeynet system which is capable of deploying both low level interaction and high level interaction simultaneously. This honeynet system provides the user with the ability to scan a network passively or actively, stores the data from the scans to create a network depiction, and creates a Honeyd configuration file for deployment of low interaction honeypots and an extensible markup language (XML) file for the deployment of high interaction honeypots. Also to reduce the burden on the user to constantly supervise the individual processes, two management programs were created to oversee the modules and control the creation of the honeypot configuration files. The active and passive scanning modules have been divided into two respective management programs. The passive network scanning module is managed by *honeypot_scanner* and active is managed by *active_scanner*. Also a single database design has been implemented to store the information necessary for the system's operation and data gathered during scanning. For the low interaction honeypot, *honeypot_scanner* is creating the Honeyd configuration files, information is being stored into two tables and another table is being queried. These tables keep track for information which enable the low interaction honeypot to be used to their fullest potential. But, for high interaction honeypots, tables are not used in the system, the tables allow for an expanded design which incorporates and deploys both low and high interaction honeypots. The results demonstrated that both passive and active scanning can be used simultaneously to gather an accurate picture of the network environment. Extensive information can be gathered to create a honeynet which is representative of the production environment. POf was able to recognize the Windows operating systems although the ability was degraded when active scanning was included. Xprobe2 had great success at identifying the Linux operating systems. Nmap and tcpdump were able to observe a majority of the ports which were open and communicating. The combination of all the scanners allowed for a more complete picture to be obtained. The

author also targets the organizations by describing that due to the nature of the system, organizations are hesitant to allow the installation of a device by a researcher which captures packets on their corporate network. However, when the system is deployed by the organization then tests can be conducted to determine the optimal noise level for their environment.

Another Honeypot which is Web-based deployed by Nisarg Thakur and other co-members [3] have proposed a low interaction web based honeypot to bite the attackers. The honeypot would not only record the attacker's request, but also try to expose the attacker's identity at the same time and prevent any further attacks from the same source by blocking its IP or MAC address and locating him geographically. The proposed system classifies the user by the request it sends to the server. If the user is a normal user it will be served normally. If it is an attacker, then send an emulated webpage pretending to be a real webpage. The webpage serves the attackers attack as to give the attackers a false conception that the attack is successful by sending the desired output with attacked JavaScript. The Js runs on the attacker's browsers and sends the information like IP, MAC address, and attacker‟s social media account credentials. The information obtained is logged and the IP address is blocked for the further prevention of the attack. Also by using software‟s like GeoPlot and the information the attacker‟s location is plotted geographically. The proposed system's is divided into three main modules:

- Classification of Request.
- Emulator to serve attacker requests and further get its information.
- Maintaining the database of list of attackers and tracking their geographical location Classification of Request is needed to differentiate the attacker and the genuine user. It is done by inspecting the query entered by the user. Honeypot accepts the query, assesses it and classifies it as the attacker‟s query or genuine request. Whenever the query is classified as Attacker‟s query, the attacker‟s request is then forwarded to the emulator. To lure the attackers, XSS and SQL Injection vulnerabilities are emulated so that they will think that the web -page is vulnerable. XSS and SQL Injection attacks are chosen since they are the most conducted attacks nowadays. A fake interface was designed in order to attract attackers and make them think that the website is vulnerable. They have targeted real life attackers who opened the website, not bots nor machines. Overall the fake page is just designed as the real web-page as the attacker must not suspect that he is attacking a fake page not the real web-site. The main page only consists of several fake information and obfuscated JavaScript code. With this, we made the attacker think that our honeypot was an institution news website. To get the attacker‟s information, JavaScript is utilized. The proposed honeypot makes use of the LikeJacking technique which is usually used by black-hat advertisers. In LikeJacking, this dummy Facebook page is liked by the attacker accidentally when they visit the honeypot. If the facebook method fails then the IP address or MAC address would be captured. The information retrieved back from the JavaScript is stored in a Database so that the attack from the same source can be prevented again. The database is a simple MySQL database which has information such as IP address, MAC address and Source Country and if available Facebook account User name and User ID. The Geographical location of the attacker is found by the source IP address sent by the Java Script is forwarded to the WHOIS database to find the origin country.

JR Kondra has proposed a new approach as compared to the existing shortcomings in the security scenario [4]. It uses the virtualization technique to overcome the existing security problem. It overcomes the limitation of honeypots from single network detection to network across the organization and improves the existing security design to waste the attackers' time as much as possible to get the best useful information. The proposed approach collaborates the concept of Honeynet, honeyd and honeypots related security resources. Honeyd is a low-interaction honeypot which can detect and also log any activity on any port (UDP or TCP), and also for some ICMP ports. Honeyd must be configured with attack signatures so that it can recognize the type of attacks. Honeyd has the capability to interact with the attackers. Therefore, Address Resolution Protocol Daemon (ARPD) is required in order to detect in the first place that there is someone who is trying or requesting to interact with a nonexistent host. ARPD is a software that actually monitors the unused IP space and directs attacks to the Honeyd honeypot. Snort is also used as an intrusion detection, it has real time alerting capability and generates an alarm of each incoming and outgoing packet. If a malicious packet is found, then snort generates a real-time alarm and all the suspicious connections are forwarded through the security resources. The information gathered from the analysis with the help of different analysis tools used to extract the possible information about the attacker. Logs generated were stored on the server and analysis tools were used for analyzing the logged activities.

Vivekanand Rajbhar has developed a framework of honeypot which is designed for windows 64 bit operating system [5].The idea is to develop a java based portable honeypot that has IDS & IPS embedded within itself. Proposed honeypot captures packets in real time using Jnetpcap,Winpcap powershell libraries and stores all the

packet data into an embedded Jderby database. Real time IDS is implemented in honeypot by using JPowershell, IDSrules and algorithms. While IPS is implemented by using Jpowershell, custom rules & windows default firewall. The proposed system differs from traditional honeypots as it is a single instance multithreaded java based portable honeypot which uses custom JnetpcapAPI with Winpcap and Jpowershell to capture and fetch packet information. But instead of creating a TCP dump file it uses an embedded Jderby database to store all the packet information, which enables this honeypot to implement a real time intrusion detection system within it. When intrusions happen, administrators can blacklist it to prevent any further intrusions from the same source. As soon as the admin adds IPS rules in the honeypot a new rule is automatically generated in the firewall which immediately starts preventing intrusions from the network. Logs and reports are also generatedfrom whole process which are stored in a database for further analysis.Thus, this honeypot implements real time IDS & IPS which improves the effectiveness of honeypot in network security.

Vaishali Shirsath has conducted a survey on the current states of Honeypot [6]. She has also reviewed the problems related to honeypot and deception based defensive strategies inside the cyberworld, this paper gives overview of honeypot techniques and various types of honeypots and the different deception techniques used for counter assaults. Some famous honeypots mentioned in this paper are Spam honeypot: conjointly called as spam trap. Malware honeypot: This kind of honeypot is made to recreate powerless applications.Database honeypot: Databases are a standard objective of web assailants, and by setting up a database honeypot one can watch and learn diverse assault procedures like SQL infusion, benefit misuse, SQL service abuse and undeniably more. Spider honeypot: This kind of honeypot works by making bogus sites and connections that are exclusively open by web-crawlers, not by people. While tending to honeypot problems, one can partition it into two fundamental regions. The essential territory is the advancement of the honeypot, its productive organization, and economical upkeep. The subsequent territory is the examination of gathered data, its representation, information extraction and higher subjective procedure upheld the information, while entirely unexpected service and honeypot types face various issues in these two territories, during this paper, abridge the general issue that emerge in the vast majority of the honeypot examples, with accentuation on the momentum condition of the honeypot look into. Challenges like a) Challenges to Develop Honeypot b) Challenges to Scale Honeypot. To conclude no tool can be great and perfect, security is scarcely accomplished with the blend of all, despite the fact that deception can provide us with significant data about the assault and how to forestall it later on, it can't stop the assault itself, honeypots can be one of the developing computer security innovations. The primary thought behind the honeypot is utilizing the duplicity to assemble the information regarding the assailant's exercises and strategy. There's also another survey conducted by Yamini Shegaonkar about implementing multi-level security using honeypot [8]. They have used a king protea to form a real-world situation. The king protea could be a well-designed system that pulls hackers. By attracting hackers to your system, you'll be able to monitor the processes that hackers begin and run on your system. That is, the king protea could be a lure machine that appears sort of a real system to draw in attackers. the aim of honeypots is to investigate, understand, observe and track hacker behavior so as to form a safer system. king protea could be a great way to enhance the information of network security directors and learn the way to use rhetorical tools to urge info from the victim's system. Honeypots also are terribly helpful for future threats that may track attacks from new technologies. In this paper they have taken into account the latest advances in Honeypot. Some remarkable suggestions and analysis were discussed. Aspects of the use of Honeypot in the formation and in the hybrid environment with IDs were explained. In this article, authors also define the use of signature techniques in Honeypot for the traffic analysis. Paper also proposes a methodology for design and implementation of honeypot. In this paper the working of honeypot has been studied and to interact with the attackers and malwares.

III. Approach

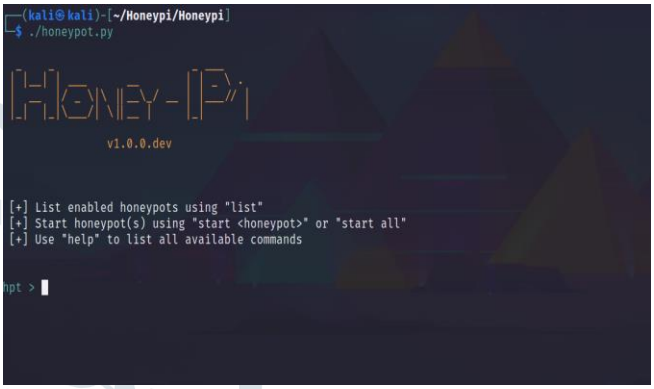
The Honey-Pi project works by emulating a web server and logging incoming TCP connections that attempt to send data to the emulated server. By doing so, it can capture information about potential attackers and the methods they are using to try and compromise the server. The program uses Python's asyncio library to manage multiple concurrent connections, and it can be configured to emulate various web server software, such as Apache or NGINX. Additionally, it provides logging functionality that records details about incoming connections, such as the source IP address, the user agent string, and the number of bytes sent. Honey-Pi is a honeypot "platform" for tracking and monitoring UDP-based Distributed Denial of Service (DDoS) attacks. The platform currently supports following honeypot services/servers in form of relatively simple plugins called pots:

a) DNS server:

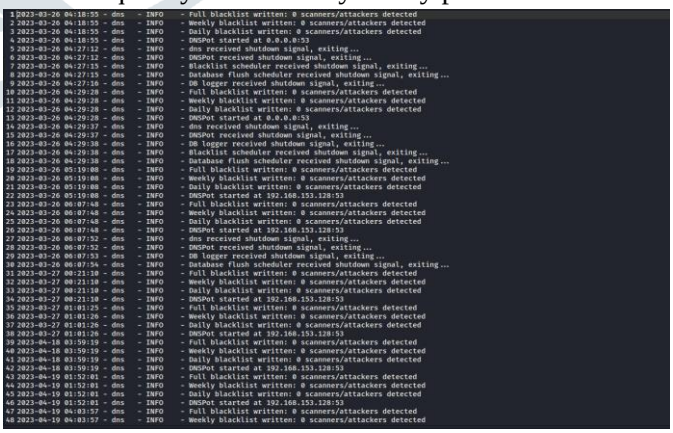
The DNS plugin is based on the UDPot project and uses the Twisted framework/engine to implement a simple DNS server that can be used to generate DDoS traffic. The plugin attempts to emulate a real DNS service as closely as possible by forwarding all incoming DNS requests to a valid recursive resolver, and returning arbitrary responses to certain queries. When a client sends a DNS query to the emulated DNS server, the plugin generates a response and sends it back to the client. The size of the response depends on the type of query and the data contained in the query. If the response is larger than the original query, this can potentially be used to amplify the size of a DDoS attack. To limit the impact of the DNS plugin on the wider internet, the plugin is configured to forward all requests to a valid recursive resolver and only returns arbitrary responses to CHAOS queries. This helps to prevent the plugin from causing any actual harm to legitimate DNS traffic. Overall, the DNS plugin in the ddospot project provides a way to generate DDoS traffic that targets DNS servers, which can be used to test the resilience of DNS infrastructure and to develop and test mitigation strategies. However, it is important to use the plugin responsibly and with caution to avoid causing any harm to legitimate network traffic.

Generic server:

The project includes a plugin called the "generic" plugin, which can be used to emulate the behavior of arbitrary network services without following specific protocol specifications or algorithms. The plugin allows users to define a fixed response that will be returned for every incoming query, or to generate a random response that varies in size depending on the size of the incoming query. This can be useful for simulating different types of network traffic and testing the resilience of network infrastructure against DDoS attacks. One potential use case for the generic plugin is to generate DDoS traffic that targets a specific service or network resource. By emulating the behavior of a particular service and generating traffic with a high amplification factor, an attacker could potentially overwhelm the target network and cause it to become unavailable. It's worth noting that while the generic plugin can be a useful tool for testing and analyzing network infrastructure, it can also be used for malicious purposes. As with any tool that can be used for generating DDoS traffic, it's important to use the generic plugin responsibly and with caution to avoid causing harm to legitimate network traffic. The project provides a command-line interface that allows users to interact with the honeypot and perform various tasks such as starting the honeypot, checking its status, and stopping the service. Users can use the command-line interface to start the honeypot by running the ddospot command followed by the desired options and parameters. Once the honeypot is running, users can use the status command to check its current status and see if any incoming traffic has been detected. In addition, users can use the stop command to stop the honeypot and shut down the service. This can be useful for temporarily disabling the honeypot or for shutting it down completely after testing is complete. Overall, the command-line interface provides a simple and convenient way for users to interact with the honeypot and perform various tasks. By using the interface, users can easily start, monitor, and stop the honeypot as needed, making it a valuable tool for testing and analyzing network infrastructure.

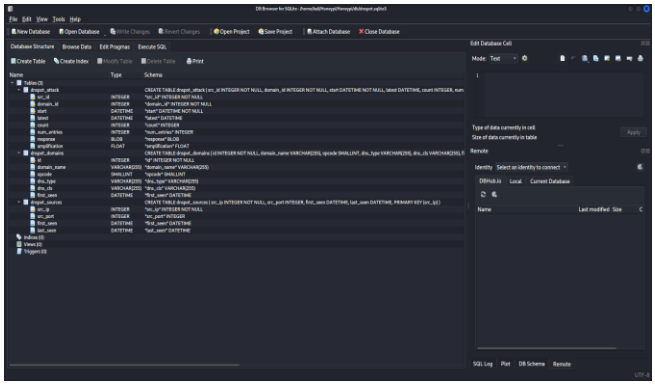


The project is also designed to collect data on potential attacks through the use of a log file and an SQL database. The honeypot logs incoming traffic, including the IP addresses of the sources, the number of packets received, and other relevant information. The log file is a text file that contains a detailed record of all incoming traffic to the honeypot. It includes information such as the source IP address, the protocol used, the timestamp of the request, and the number of packets received. This information can be used to analyze the nature and origin of the traffic, as well as the frequency and intensity of any potential attacks.



In addition to the log file, the honeypot also uses an SQL database to store and manage the collected data. The database provides a more structured and organized way of storing and querying the data, allowing for more efficient analysis and reporting. The SQL database is designed to collect and store a wide range of information, including the source IP address, the type of attack, the size of the packets

received, and the frequency of the attacks. This information can be queried and analyzed to identify patterns and trends in the attacks, as well as to develop countermeasures to prevent future attacks.



Overall, the combination of the log file and the SQL database provides a powerful tool for analyzing potential attacks on the honeypot. By collecting and storing detailed information on incoming traffic, the honeypot can help organizations identify potential security threats and develop effective strategies to protect against them.

IV. Conclusion

Setting up a honeypot can be an effective way to protect your organization from cyber attacks. Not only will it waste a hacker's time, but it will also provide valuable insights into the methods used to attack your system. By monitoring a honeypot, you can gain a clear understanding of the types of attacks being used and the techniques hackers employ. This information can then be used to improve your organization's defenses, ensuring that your real systems and data are well protected against future attacks. Security researchers have long recognized the benefits of honeypots, which have been critical in the study of hacker behavior. By using honeypots to monitor and analyze attacks, researchers can identify new attack vectors and develop countermeasures to prevent them. Overall, honeypots are a valuable tool in any organization's security arsenal. They not only help protect against attacks but also provide critical insights into the methods and motivations of attackers. By using honeypots, organizations can stay one step ahead of potential threats and keep their systems and data secure.

However, it's important to note that using honeypots for this purpose can be a legally gray area, and caution should be exercised to avoid accidentally attracting legitimate traffic to the honeypot server. Additionally, the use of honeypots should be in compliance with applicable laws and regulations.

V. Future Scope

The future scope of the project includes configuring it to detect more types of attacks. As new attack vectors emerge, the honeypot will need to be updated to recognize and respond to these threats. This could involve adding new plugins or modifying existing ones to detect and mitigate different types of attacks. Another future scope for the project is to integrate it with an Intrusion Detection System (IDS) and Intrusion Prevention System (IPS) to provide complete security. By combining the honeypot with an IDS/IPS, organizations can detect and block potential threats in real-time, reducing the risk of a successful attack. The project could also be expanded by adding extra honeypots with various functions. For example, honeypots could be set up to mimic different types of servers or services, such as email servers or file transfer protocols. This would allow for a more comprehensive view of potential threats and provide greater insight into the types of attacks being used. Finally, the project could benefit from a dedicated server for increased performance and power. With a dedicated server, the honeypot could process more traffic and respond more quickly to potential threats. This would make it easier to identify and mitigate attacks, improving the overall security posture of the organization.

VI. References

[1]. Tiwari, Aparna, and Dinesh Kumar. "Comparative study of various honeypot tools on the basis of their classification & features." *Proceedings of the International Conference on Innovative Computing & Communications (ICICC)*. 2020.

[2]. Hecker, Christopher R. *A methodology for intelligent honeypot deployment and active engagement of attackers*. Diss. 2012.

[3]. Nisarg N Thakur, Prashant Patil, Rajat Varade, and Abhishek Pawar. "Web-based honeypot for detecting and tracking attackers." *International Journal Of Advance Research And Innovative Ideas In Education* 2.3(2016) : 3273-3277.

[4]. Kondra, Janardhan Reddy, et al. "Honeypot-based intrusion detection system: A performance analysis." *2016 3rd international conference on computing for sustainable global development (INDIACom)*. IEEE, 2016.

[5]. Rajbhar, Vivekanand. "INTRUSION DETECTION & PREVENTION USING HONEYPOT." *International Journal of Advanced Research in Computer Science* 9.4 (2018).

[6]. Shirsath, Vaishali. "A Survey on Current States of Honeypots and Deception Techniques for Attack Capture." *International Journal of Engineering Research & Technology* 9 (2021): 438-443.

[7]. Atashzar, Hasty, et al. "A survey on web application vulnerabilities and countermeasures." *2011 6th International Conference on Computer Sciences and Convergence Information Technology (ICCIT)*. IEEE, 2011.

[8]. Shegaonkar, Yamini S., Leena Patil, and Shrikant Zade. "Survey on Multilevel Security Using Honeypot." *IJISRT*.

International Journal of Innovative Science and Research Technology 6.5 (2021): 959-963.
[9]. Hoepers, Cristine, et al. "A national early warning capability based on a network of distributed honeypots." *17th Annual FIRST Conference on Computer Security Incident Handling, Singapore*. 2005.
[10]. Ikuomenisan, Gbenga, and Yasser Morgan. "Systematic Review of Graphical Visual Methods in Honeypot Attack Data Analysis." *Journal of Information Security* 13.4 (2022): 210-243.
[11]. Dicholkar, Supriya Vishal, and Deepthi Sekhar. "IoT Security Research Opportunities." *2020 International Conference on Convergence to Digital World-Quo Vadis (ICCDW)*. IEEE, 2020.
[12]. Pal, Prabhakar, and Vinit Mehta "Cybersecurity: Various methods and techniques of Cybercrime", *International Journal of Emerging Technologies and Innovative Research*, 2349-5162, Vol.10, Issue 1, page no.a732-a737, January-2023
[13]. Mehta, Vinit Hemanshu, and Prabhakar Pal "Internet Security - Intrusion Detection Systems ", *International Journal of Emerging Technologies and Innovative Research*, 2349-5162, Vol.10, Issue 1, page no. a751-a756, January-2023