Chapter One

Introduction

1.1 Background to the Study

The connectivity provided by the internet has dramatically affected the world in which we live. Due to the fact that practically all corporate transactions, academic research, and personal communication now use and depend on this channel, it has become important to protect oneself from attacks[1]. The apparent increase in computer incidents after 1999 as a result of automated and targeted assaults from across the world has spawned a multibillion-dollar industry dedicated to identifying and preventing unwanted penetration attempts[1]. In order to reduce the number of false positives in these Intrusion Detection System (IDS) logs, a decoy technology has been created whose sole purpose is to be attacked and compromised in order to assist the system administrator in determining the method of both penetration and execution for future attacks[2]. Since their introduction in the 1990s, honeypots systems have enabled researchers and businesses to collect tools and uncover system weaknesses. Since honeypots are decoy devices and give no production value to the environment, they offer the opportunity to monitor the network without the tedious analysis inherent to monitoring a production device[3].

Analyzing a production device forensically requires a deep dive into the many records and applications required to do business. On a non-production device, assaults may be separated from network noise and their source can be immediately identified. This efficacy and efficiency have contributed to the development of honeypot systems[4].

In information security, the most easily protected equipment is hardware: conceal it in a room, chain it to a counter, or buy an extra. Information presents extra obstacles. It may

exist in several locations, be transferred halfway over the world in a matter of seconds, and be taken without your knowing[5].

Since network-based incursions have expanded quickly over the past several years as a result of the proliferation and popularity of readily accessible attack tools, this trend is expected to continue[6]. Due to this surge in incursions, the idea of network honeypots is being created in order to capture and decode the attack tactics of malicious attackers. This project examines the present status of honeypot technology and provides a methodology for evaluating and enhancing the effectiveness of Deceptive Honeypots especially in the area of detecting fake honeypots that could be developed by the attackers[6].

Prior to the invention of computers, information security via deceit was frequently proven. This was argued in the fifth century B.C. that "all combat is founded on deceit[7]. Deception is the dissemination of false information that is plausible enough to mislead an opponent's situational awareness and to affect and misdirect his perceptions and decision-making processes[7]. It has been suggested that influencing an enemy's perspective of reality via defensive deception might possibly level the cyber battlefield[8]. While reactive detection-based defense technologies remain a core component of cyber defense, they are no longer adequate in addressing the ever-increasing threats of evolving cyber attacks[8].

The purpose of deception technology is to prevent a cybercriminal from causing major harm after infiltrating a network. The device operates by establishing traps or deception network address shuffling that imitate real infrastructure technology assets[8]. These network address shuffling techniques may operate in a virtual or actual operating system environment and are meant to deceive cybercriminals into believing they have found a means to elevate privileges and steal credentials[9]. Once a trap is sprung, alerts are sent to a

centralized deception server that logs the impacted network address shuffling and attack pathways[9].

Humans are susceptible to cognitive limits and prejudice, which may result in poor decision making and deviant conduct. While cyber attackers often exploit these shortcomings (e.g., spear phishing and spam), little research has been conducted on their use for cyber defence[10]. As network administrators, defenders might utilize their home-field advantage to provide information to attackers in precise ways to exploit and amplify fundamental human biases in order to delay, disrupt, or prevent the assault. Deception for cyber defense promotes this objective, which is to counterbalance the asymmetric character of computer defense by raising the attacker's burden and lowering the defender's via strategic interactions with the person behind the cyber-attack[11].

Honeypot is a computer system phrase that refers to a trap established to detect, reflect, or otherwise counteract any unlawful behaviour carried out by an authorized or unauthorized person or system[12]. Honeypot has its own requirements, such as a computer system, data, and a network on which that computer system will operate, which implies that computer system must be on a network in order to monitor activity on a computer system across a network[13]. Honeypot can also be explained as a system containing information whose value is contingent on the illicit use of data[14]. As network-based attacks are made by attackers, honeypots operate online[13]. The honeypot is an electrical lure. It seems to be part of the network, but it has been installed to monitor a hacker's activity[14]. Comparing honeypot and firewall reveals a reversed operating principle for both, as honeypot allows all incoming traffic to enter but prevents it from leaving, while firewall prevents illegal activity from entering a system[14]. Honeypot employs wary technology and is an optional method for

protecting the network and searching in order to construct a robust system in a descriptive setting. The described how Honeypot raises an alert to the system administrator while an attacker strikes the system and provides a wakeup call to the client to investigate the attacker's activity[15]. While the attacker is performing some activities on the system, the honeypot will identify the attacker and collect malicious activities. Additionally, it will monitor the attacker's behaviour and record his activities so that it is simple to estimate the level of the attack and to determine what tool will be necessary to prevent such activities in the future[15].

Honeypot technology was elaborately discussed, citing its utility in network security defense and the development of several honeypot-based distributed intrusion prevention models[16]. This study paper is organized into segments, and each chapter contains useful information on honeypots[16]. Using a variety of definitions, Chapter I, Introduction offers an accurate understanding of what a honeypot is. In Chapter II, the earlier efforts on honeypots are described in depth, and the honeypot system, including its operating principle and processing flow, is illuminated. In addition to discussing the Literature review, the history of honeypots, and the numerous varieties of honeypots, this part explores some models that are based on honeypots. Several uses, benefits, and drawbacks of honeypot were also explored in this chapter. While chapter III examined the methodologies to be used in conducting the study and detailed system design and analysis in depth, the instrument used for data collecting, the hardware and software requirements, and how a simulator is used to build fake accounts that are tested and run decoy deception to determine the effectiveness of honeypot technology will be examined[17]. The testing and installation of the honeypot deception technology will be the emphasis of Chapter IV in

order to determine its degree of efficiency and efficacy. This chapter will also cover results to determine the degree of trust in the implemented deception technique. The last chapter, chapter V, will provide the conclusion, recommendations, and suggestions for further study on the honeypot technology[17].

Honeypot operates as a resource that operates over a network, it is designed so that it behaves like a host, which attracts attackers; however, its main purpose is to be attacked by the attacker and explored over the network; data on it may be fake but looks real, giving the attacker the impression that it's a real host, causing more attackers to target it. Its function is to store and retrieve the activities performed by the attacker, which is possible due to the presence of software that runs in the background and logs network communication between the attacker and honeypot host[17]. Various analytical tools were used to analyze the data or activities performed by the attacker to determine the cause of the attack[18].

Honeypot is a computer system term that refers to a trap established to detect, reflect, or otherwise counteract any unlawful behaviour carried out by an authorized or unauthorized person or system[18]. Honeypot has its own requirements, such as a computer system, data, and a network on which that computer system will operate, which implies that computer system must be on a network in order to monitor activity on a computer system across a network[19]. In one of the researches, honeypot was described as a system containing information whose value is contingent on the illicit use of data[19]. As network-based attacks are made by attackers, honeypots operate online[19]. The honeypot is an electrical lure. It seems to be part of the network, but it has been installed to monitor a hacker's activity[20]. Comparing honeypot and firewall reveals a reversed operating principle for both, as honeypot allows all incoming traffic to enter but prevents it from leaving, while firewall

prevents illegal activity from entering a system[20]. Honeypot employs wary technology and is an optional method for protecting the network and searching in order to construct a robust system in a descriptive setting. The description of how honeypot raises an alert to the system administrator while an attacker strikes the system and provides a wakeup call to the client to investigate the attacker's activity[21]. While the attacker is performing some activities on the system, the honeypot will identify the attacker and collect malicious activities[21]. Additionally, it will monitor the attacker's behaviour and record his activities so that it is simple to estimate the level of the attack and to determine what tool will be necessary to prevent such activities in the future[22].

Additionally, Honeypot technology was in its utility in network security defense and the development of several honeypot-based distributed intrusion prevention models[22]. This research is organized into segments, and each chapter contains useful information on honeypots[23].

Honeypot was explained by other researchers as a resource that operates over a network; it is designed so that it behaves like a host, which attracts attackers; however, its main purpose is to be attacked by the attacker and explored over the network; data on it may be fake but looks real, giving the attacker the impression that it's a real host, causing more attackers to target it[23]. Its function is to store and retrieve the activities performed by the attacker, which is possible due to the presence of software that runs in the background and logs network communication between the attacker and honeypot host[24]. Various analytical tools were used to analyze the data or activities performed by the attacker to determine the cause of the attack[24]. However, the basic operation of a honeypot system in cyber security can be found in Appendix I.

One of the articles explained that a simple honeypot system has three distinct modules: the Induced Module, the Deceived Module, and the Analysis Module. Each of these three components has its own distinct functionality and purpose[23]. Induced module functions to lure an attacker to the honeypot system[24]. As it was already known, a honeypot system has its own database; thus, fooled calls simulation over the data in the database to produce bogus information that is given to the attacker[23]. The actions conducted by both modules are saved and evaluated by the Analysis module, which then modifies the activities performed by both modules. The diagram in appendix 1 depicts the processing flow of Honeypot system.

In network intrusion situations, an attacker only knows what they see[25]. The attacker is typically thousands of kilometers from the target network. This was explained that networks typically provide attackers more information than defenders want[24]. The network owner may communicate deceptive information with the attacker[25]. Because network information is typically confusing and incomplete, opportunity arises in chaos for dishonesty[24]. Deception may affect an attacker's thinking, self-confidence, and decision-making more than traditional defenses. It also allows a defense to manipulate an attacker's knowledge. The basic processing flow of Honeypot system can be found in Appendix II.

1.2 Statement of the Problem

The security of vital information and infrastructure has become an issue of concern among individual, corporate and government organizations. As more crucial and secret information is kept digitally, safeguarding critical infrastructure will remain a top concern in order to facilitate cyber-threat-free retrieval of information, protection of large-scale,

global infrastructures and preventing unauthorized, harmful organizations from obtaining a foothold in the system is a primary concern. The introduction of honeypots systems has ushered in a new era for the security of information on network domain systems, where deception applications are meant to imitate the genuine system in order to fool attackers who aim to hack and destroy the system. Vital information is still vulnerable and exposed to threat and attack despite the development of this innovation of honeypot deception technology to confuse the attackers. This is due to the discovery that the attackers have come up with another tactic of developing a fake deception system to cause confusion and appear like the real honeypot system designed by the developers or administrators of the system.

Many researchers have primarily focused on deceiving and luring cyber attackers through traps and decoy accounts into the system with the sole purpose of obtaining intelligent information from them, without taking into account the possibility that attackers could develop a fake honeypot system that can operate in parallel with the real honeypot system in order to confuse the security administrator.

In such circumstance, there is a need to discover a solution to the issue of attackers establishing a false honeypot system that appears like a genuine honeypot in order to confuse the system administrator, In spite of different deception technology techniques that are being deployed by cyber security experts to prevent attackers from gaining access to the cloud information, the threat increases every second as attackers continuously exploring different means to gain unauthorize access the cyber space. Hence, the need for effective, robust and more efficient models to mitigate threats poses by Cyber criminals. Hence the

need for this research that implements a web based low interaction Glastopf honeypot for Accurate gathering of Attackers information and detection of fake honeypot.

1.3 Aim and Objectives of the Study

The aim of this study is to design and optimize the accuracy of the attacker information gathering and Detect Fake Honeypot

Research Objectives are to:

I. Design a model that can enhance the performance of existing Glastopf honeypot to accurately detect the fake honeypot introduced by the attackers in to the system
II. Implement the designed model on a Web Application
III. Test the effectiveness of the enhanced Glastopf honeypot system
IV. Evaluate the performance of enhanced Glastopf in (3) against the existing honeypot system

1.4 Significance of the Study

I. This study will help security experts to gain more knowledge about deception technology
II. It will assist in building a robust database of the activities of cyber criminals
III. It will also provide adequate digital information about cyber criminals to the information security experts for the purpose of forensic.
IV. It will expose Cyber Criminals secret methods of deploying fake honeypot system to deceive the Security Administrator
V. It will assist System Administration to understand behavioural pattern and source of cyber criminals

1.5 Scope of the Study

The scope of this work is basically to simulate of virtual honeypot system that would guide against cyber information from being maliciously attacked by the cyber criminals. The study is expected for completion with twelve months after the gathering of necessary information needed. The following are to tools and dependencies required for the completion of the study.

I. This research obtained an open-source dataset (Hornet40-traffic-perhoneypot-h) from Mendeley Data repository.

II. Glastopf honeypot was used due to its low interaction system and operate within a Web application that makes it suitable in any environment

III. It utilized OMNET++ Simulation tool for the Design of Virtual Machine.

IV. Performance evaluation of the model is computed and compared.

V. Output is compared with existing works.

Although there are some other instances like when an attacker tries to design fake honeypot system that can work in parallel with the real honeypot with the purpose of creating confusion for the administrator. Further research can be conducted on that aspect to see if honeypot has the ability to detect fake honeypot developed by the attacker and how it can be mitigated.

1.6 Limitations of the Study

Due to the fact that there is a limited number of resources needed for the execution of this research, it would require more efforts and dedication to put in place in making sure that the limited resources are judiciously utilized.

Other various limitations especially to the honeypot research work are highlighted bellow.

1 The security tests of the Glatopf honeypot model are confined to a given number of vulnerability classes due to the web-based attacks, while security testing process is needed to be conducted through both low-level Since low-level interaction honeypot has been used, the high-level interaction honeypot is less likely to be known for the vulnerability assessment.

2 The automated tests of security can last for a given number of periods.

3 The vulnerability scanner has the potential to provide a true negative or false-positive result.

1.7 Operational Definition of Terms

1. Computational Method: This is a study technique that employs new breakthroughs in computing, such as algorithms, models, simulations, and systems, to comprehend intricate social, biological, technical, and countless other patterns and behaviours.

2. Malicious: These are activities that are designed to compromise the confidentiality, availability, or integrity of computers, information or communications systems, networks, physical or virtual infrastructure managed by computers or information systems, or data located thereon.

3. Decoy Account: An account that is established to determine whether someone is trying to log in. When an attempt is made, security specialists may analyze the approaches and strategies of the attackers without being caught or compromising any data.

4. Deception Technology: is a cyber-security defensive approach that tries to deceive attackers by simulating legitimate assets using traps and decoys.

5. **Intrusion Detection System (IDS):** It is a monitoring system that detects suspicious activities and generates alerts when they are detected. On the basis of these warnings, a Security Operations Center (SOC) analyst or incident responder may examine the situation and implement the necessary measures to eliminate the danger.

6. **Intrusion Prevention System (IPS):** A network security instrument (which might be a physical device or software) that continually monitors a network for harmful behaviour and prevents it by reporting, blocking, or dropping it when it occurs.

7. **Honeypot System:** A security technique that constructs a simulated trap to entice intruders. A maliciously infiltrated computer system enables attackers to exploit weaknesses, allowing you to research them and enhance your security measures.

8. **Honeynet:** a network designed to lure and divert prospective attackers from your production network. In a honeynet, attackers will discover not only susceptible services or servers, but also vulnerable routers, firewalls, and other network border devices, security apps, etc.

9. **Cybersecurity:** Cybersecurity is the use of technology, processes, and policies to protect systems, networks, programs, devices, and data against cyber-attacks. It attempts to reduce the likelihood of cyber-attacks and prevent the illegal use of systems, networks, and technology.

10. **Firewall:** A network security device that monitors and filters incoming and outgoing network traffic according on a company's predefined security criteria. At its most basic level, a firewall is the barrier between a private internal network and the Internet.

11. **Cloud Document:** This is an open-source software as a service platform that allows users to upload, analyze, annotate, collaborate, and publish primary source documents.

12. Omnet++ is an extensible, modular, component-based C++ simulation toolkit and framework that is mainly used for the development of network simulators.

13. Cloud Computing is a technology that uses the internet to store and manage data on remote servers, and then accesses the data over the internet. This kind of apparatus permits remote operation. Cloud computing customers do not own the real infrastructure; they rent the service from a third party[25].

14. Computer Framework: A computer framework is often a layered structure that describes the sorts of program that may or should be developed and their interrelationships. Some frameworks for computer systems include actual program, describe programming interfaces, or offer programming tools for their execution[26].

15. Cyber Deceptions: This advanced cybercrime, also known as Spear-Phishing or Social Engineering, is meant to get access to your firm's finances by overcoming its current protections.

16. Network Security: is a group of technologies that protects the functionality and integrity of an organization's infrastructure by blocking the entry or spread of a variety of potential threats inside a network[23].

17. Cyber Terrorism: The term "terrorism" may refer to the illegal use of force or cruelty against persons to threaten an administration or its citizens and associations in order to accomplish a political or malicious objective.

18. Hacking: is the broad term for any unauthorized network access to any "computer system" and is assessed as "cyber murder" for any building. The bulk of these cybercriminals use "brute force," which is the combination of every conceivable letter, number, and image until they get the password[26].

Endnotes

[1]A. Abdulrahman, M. Ishaq, A. Fatima, A. Atika & Y. Suberu. "*A Proposed Improved Captcha Based Intrusion Detection Model*"**, Journal of Advanced Science and Optimization Research** Vol. 27, No.9, 2023 ISSN 2418-9325

[2]A. Ahmim, L. Maglaras, M. Ferrag, M. Derdour & H. Janicke. "*A Novel Hierarchical Intrusion Detection System Based on Decision Trees and Rules-based Models*". In 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), 2019, 228-233, DOI: 10.1109/DCOSS.2019.00059

[3]A. Arkhipova & D. Karevskiy. *"Honeypot as a Tool for Creating an Effective Secure System".* Novosibirsk State Technical University in Digital Technology Security Digital Technology Security 2021 ; https://doi.org/10.17212/2782-2230-2021-2-122-135

[4]A. Christin, C. Giselle, A. Wesam, A. Abu & S. Maha. "*A Comprehensive Survey for IoT Security Datasets Taxonomy, Classification and Machine Learning Mechanisms.* **Computers & Security,** 2023, 103283, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2023.103283. **(https://www.sciencedirect.com/science/article/pii/S0167404823001931)**

[5]A. Elkosairy & A. Marianne. "*A New Web Deception System Framework"*, Conference: 2018 1st International Conference on Computer Applications & Information Security (ICCAIS) 2018, DOI: 10.1109/CAIS.8442027.

[6]A. Mishra & Sanjay K. Jain. "*A Survey on Question Answering Systems with Classification*". **Journal of King Saud University-Computer and Information Sciences** 28.3 2016, pp. 345–361. https://doi.org/10.1016/j.jksuci.2014.10.007

[7]A. Mudgal & S. Bhatia. **"***Spark-Based Network Security Honeypot System: Detailed Performance Analysis***"** Article, Dec 2022, **International Journal of Safety and Security Engineering.** 12. 2022, 737-743. 10.18280/ijsse.120610.

[8]A. Pashaei, Mohammad E. Akbari, Mina Z. Lighvan & C. Asghar. *"Early Intrusion Detection System using Honeypot for Industrial Control Networks"*, Results in Engineering, Volume 16, 2022, 100576, ISSN 2590-1230, https://doi.org/10.1016/j.rineng.2022.100576.

[9]A. Riancho. W3AF USER GUIDE. Available at: URL: http://cyber. lockheedmartin. com/hubfs/Gaining the Advantage Cyber Kill Chain. 26, 2021.

[9]A. Schlenker, O. Thakoor, H. Xu, F. Fang, M. Tambe, L. Tran-Thanh, P. Vayanos & Y. Vorobeychik. "*Deceiving Cyber Adversaries: A Game Theoretic Approach* 17th International Conference on Autonomous Agents and Multiagent Systems, AAMAS Volume: 2, 2018 pp. 892–900.

[10]A. Shah. "*Evaluating Network Forensics Applying Advanced Tools*". **International Journal of Advanced Engineering, Management and Science**, Vol 9 No 4 2023, http://journal-repository.theshillonga.com/index.php/ijaems/article/view/6178

[11]A. Waqas, A. Muhammad, N. Sabreena & W. Farhana, "*Detection and Analysis of Active Attacks using Honeypot*". **International Journal of Computer Applications** (0975 – 8887) Volume 184 – No. 50, 2023 IJCATM: www.ijcaonline.org

[12]A. Yaser. "*Improving Intrusion Detection Systems Using Artificial Neural Networks*". **ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal**, Vol. 7 No. 1 2018 https://doi.org/10.14201/ADCAIJ2018714965

[13]B. Gupta & A. Gupta. "*Assessment of Honeypots: Issues, Challenges and Future Directions*". **International Journal of Cloud Applications and Computing (IJCAC)** 8(1) 2018 |Pp 34 DOI: 10.4018/IJCAC.2018010102

[14]B. Mphago, & S. Mpoeleng. "*Deception in Web Application Honeypots: Case of Glastopf*". **International Journal of Cyber-Security and Digital Forensics**, 6(4), 2017, pp. 179-185, DOI: 10.17781/P002304

[15]B. Paul & M. Rao. "*Zero-Trust Model for Smart Manufacturing Industry*". **Applied Sciences Journal.** 13(1) 2023 :221. https://doi.org/10.3390/app13010221

[16]B. Sara, C. Mauro, P. Luca & P. Pier. "*Social Honeypot for Humans: Luring People through Self-Managed Instagram Pages*". **Journal of Social and Information Networks, cs.SI), Artificial Intelligence (cs.AI), Cryptography and Security (cs.CR)** 2023 https://doi.org/10.48550/arXiv.2303.17946

[17]B. Temmie, V. Andrew, J. Kimberly, W. Ferguson, B. Sara, F. Daniel. & E. Kristin. "*The Moonraker Study: An Experimental Evaluation of Host-Based Deception*". In Hawaii International Conference on System Sciences (HICSS), Maui, Hawaii, 2020, DOI:10.24251/HICSS.2020.231.

[18]B. Abbaschian, S. Daniel & A. Elmaghraby. "*Deep Learning Techniques for Speech Emotion Recognition, from Databases to Models*", Computer Science and Engineering Department, University of Louisville, Louisville, KY 40292, USA, 2021, 21(4), 1249; https://doi.org/10.3390/s21041249

[19]C. Chou, C. Wu, K. Lu, L. Hsien & J. Li. *"Modbus Packet Analysis and Attack Mode for SCADA System*" **Journal of ICT, Design, Engineering and Technological Science**. 2018, 30-35. 10.33150/JITDETS-2.2.1.

[20]C. Kai, W. Zhan, L. Dongkun & R. Mu. *"The TaintDroid Based Honeypot Monitoring System for Embedded Device"*, **Journal of Physics Conference Series** 2203 (1):012077, 2022, DOI: 10.1088/1742-6596/2203/1/012077.

[21]C. Kai, W. Zhan, Z. Chengcheng & M. Haohua. *"The Research on Network Function Virtualization Based Network Honeypot"*, Proceedings of the 12th International Conference on Computer Engineering and Networks, 2022, DOI: 10.1007/978-981-19-6901-0_156

[22]C. Kuan, L. I-Hsien & J. Li. *"Honeypot System of SCADA Security Survey",* Proceedings of International Conference on Artificial Life and Robotics 23:2018 pp 444-447, DOI: 10.5954/ICAROB.2018.OS8-8

[23]C. Sakama, M. Caminada, & A. Herzig. "*A Formal Account of Dishonesty*," **Logic Journal of IGPL,** vol. 23, 2015, no. 2, pp. 259–294, https://doi.org/10.1093/jigpal/jzu043

[24]D. Akshat, B. Anchit, A. Nihal & D. Sumithra. *"HONEYPOT: Intrusion Detection System"* **International Journal of Education Science Technology and Engineering** 3(1): 2020 pp 13-18, DOI: 10.36079/lamintang.ijeste-0301.66

[25]D. Danilov, T. Ovasapyan, D. Ivanov, A. Konoplev & D. Moskvin. *"Generation of Synthetic Data for Honeypot Systems Using Deep Learning Methods"*, Automatic Control and Computer Sciences, 2023, 56(8):916-926, DOI: 10.3103/S014641162208003X

Chapter Two

Literature Review

2.1 Conceptual Review

2.1.1 Cyber Security

A conceptual review in cybersecurity focuses on the theoretical frameworks, models, and concepts that inform our understanding of the field. It involves analyzing and synthesizing existing literature to develop a comprehensive understanding of key concepts and their applications. Here are some key conceptual areas in cybersecurity:

Threat Landscape: Understanding the threat landscape is essential in cybersecurity. This conceptual area involves studying the actors and organizations behind cyber threats, their motivations, capabilities, and techniques. It also includes analyzing the evolving nature of threats, such as the emergence of advanced persistent threats (APTs), nation-state sponsored attacks, and insider threats.

Risk Management: Conceptualizing and implementing effective risk management strategies is crucial for organizations. This involves identifying and assessing vulnerabilities, potential threats, and the potential impact of cyber incidents. It also includes understanding risk tolerance, prioritizing resources, and applying risk mitigation measures, such as implementing controls and security frameworks like ISO 27001 or NIST Cybersecurity Framework.

Defense-in-depth: The defense-in-depth model is a conceptual framework that emphasizes multiple layers of defense to protect against cyber threats. It involves deploying various security controls and measures at different levels, including network, system, application,

and user levels. This approach aims to create redundancy and resilience, minimizing the impact of a single security failure.

Incident response: Incident response is a critical aspect of cybersecurity, involving a systematic approach to manage and mitigate the impact of cyber incidents. This conceptual area encompasses developing incident response plans, defining roles and responsibilities, establishing communication channels, and conducting post-incident analysis to improve future incident response capabilities.

Security awareness and training: Recognizing the human element in cybersecurity, this conceptual area focuses on promoting security awareness and providing training to users. It involves educating employees on best practices, policies, and procedures to protect against social engineering, phishing attacks, and other human-centric vulnerabilities.

Secure software development: Conceptualizing secure software development involves integrating security measures throughout the software development lifecycle (SDLC). This includes employing secure coding practices, conducting security testing, and implementing secure configurations to avoid common security vulnerabilities and ensure the resilience of software applications.

Privacy and legal considerations: This conceptual area revolves around the legal and ethical dimensions of cybersecurity. It includes understanding privacy regulations and compliance requirements, such as the General Data Protection Regulation (GDPR) and developing strategies to protect sensitive data and ensure compliance with relevant laws and regulations. A conceptual review in cybersecurity explores these and other key areas to develop a comprehensive understanding of the theoretical foundations that underpin the

field. It helps researchers, practitioners, and policymakers advance their knowledge and develop effective strategies and solutions for addressing cyber threats and challenges.

There are two major ways that require formidable participants in security settings. The one who attempts to destroy systems and networks and the one who attempts to defend them[1]. This is formerly known as the blackhats and whitehats, and is an extreme viewpoint held by individuals following a military structure[1]. In this circumstance, it is impossible to overlook the significance of the Honeynet Project, which began in the early 90's[1]. It is, by and large, the only legitimate honeypot research organization that exists today[1]. And it is mostly accurate to believe that a global perspective on security is the proper one. However, the argument may not work commercially[1]. At this particular intersection, it is worthy to examine these two opposing viewpoints. They both need security to secure their own resources, but their techniques are distinct. It is also necessary to compare these concepts to see whether the installation of honeypots is feasible for the security sector and security research[2].

Due to the fact that almost all the gadgets are linked to a network in some capacity nowadays, network security has become very crucial[3]. Firewalls, Intrusion Detection Systems, and Anti-Viruses are used to prevent any unwanted and irregular actions on the devices, as well as the loss and abuse of data[3]. Honeypots are one such technology that may be utilized to secure device and data further[3]. By continuously monitoring the invaders' activity, it may either prevent attacks or provide information about the attackers[3].

Honeypots use a specially configured computer to detect malicious network activities[4]. A Honeypot's goal is to identify and learn from attacks in order to improve security[4]. A network administrator gains firsthand knowledge of the current threats to his network[5].

Honeypots defend against vulnerabilities that have yet to be discovered[5]. Network monitoring technologies have previously included passive devices such as Intrusion Detection Systems (IDS). IDS scan network traffic for malicious connections using patterns[6]. These might be particular words in packet payloads or sequences of packets. However, false positive alerts due to mismatched patterns are possible, as are false negative signals for legitimate attacks[7]. On a Honeypot, each and every package is suspect[7]. In a Honeypot situation, this is because the Honeypot is not registered to any production system. Regular production systems should be unaware of the Honeypot's existence[7]. Additionally, the Honeypot should not offer accurate production information. This assures that no trustworthy devices are linked to the Honeypot[8]. Consequently, every device that establishes a connection to a honeypot is either misconfigured or an attack source which makes it simple to identify honeypot assaults[8].

The function of a Honeypot determines its worth, it maximizes the value of a solution, its objective must be described as precisely as feasible. This research work presents a conceptual framework created in guidelines that are implemented with a clear statement of purpose and intended use will save time during analysis[9]. This method prohibits omitting useful findings from recorded data due to a lack of analysis strategy[9].

2.1.2 Required Features in Cyber Security

This section explains Honeypot characteristics. The paragraphs explain what should be done, but not how, to build a deployment's conceptual framework.

2.1.2.1 Security

Honeypots must never contact manufacturing machinery, a compromised honeypot might allow the theft of important data and equipment and should only communicate with the

system that infected it, reducing its value to attackers[10]. Honeypots require outbound connections to monitor and learn about new risks. A production Honeypot with little interactivity may be set up without external access. This arrangement's hazard is overlooked[10]. Different arrangements provide different risks[11]. Implementers must consider this while creating configurations[12]. It's advisable to prevent Honeypot access to production devices and leave communication channels open to the public network[12]. Also, public network access must be limited. Without this limitation, the Honeypot may be used to launch DoS/DDoS assaults or store stolen software. Restricting external connections meets this criterion[12]. Complex systems, such as an Intrusion Protection System, may filter individual flows that match specific patterns, inhibiting worm propagation and isolating sophisticated attacks[12].

2.1.2.2 Honeypot Cloaking

Connect the monitoring device to a hub between honeypot and the network to record all traffic. This blocks outgoing Honeypot traffic with passive observation and no control result[13]. Honeynet creator thinks that every packet must be scanned, then delete or transmit it[13]. A firewall is needed and it uses Layer 3, time-to-live, MAC address, and checksum are recreated during transmission. An expert burglar might reveal these changes and fingerprint the Honeywall, making the Honeypot undesirable or assaulting it[13].

2.1.2.3 Analyzability

Honeypot data reveals scans, intrusion attempts, and worm proliferation. After analyzing the packets, the investigator confronts a mound of data. Data and traffic must be separated in a way to Easily trace TCP, which Sequence numbers determine each packet flow[14]. Flowing packets reduces analysis that involves travel which made classification hard.

Destination port shows flow's intent. Flow to a particular port may not include legitimate data or be a port scan[14]. Port numbers don't always classify flows[9]. Stateless UDP connections have no flow option. When the IP protocol, source IP, source port, destination IP, and destination port are the same, IDS Snort considers a flow unique. IPv4's "Protocol" field specifies the next-level protocol. Most analytic programmes use TCP and UDP[14].

2.1.2.4 Accessibility

Well-designed systems increase data security and analysis. An operator requires speedy data access and events must be notified. Physical or network access is needed to verify data and logs often. Installations enable direct console access and tools for speedy data analysis[16]. In a hosted environment without direct access, the monitoring device should offer a data interface. Accessing the monitor increases traffic, which might disclose the Honeypot. The analytical interface must thus be accessed differently[16]. The link should also be encrypted so that its real purpose isn't revealed[17].

2.1.2.5 Alerting

Attacks need automated operator notification[18]. An intrusion-detection alerting function should deliver notifications. Sending alerts should be flexible. A backup destination route should be offered for alerting messages[18]. Outbound data should be utilized to trigger alerts instead of inbound data[19]. This includes stopping the Browser service on Windows and every other browsing tools which delivers outgoing traffic without user/hacker intervention. Outbound triggers may overflow mailboxes[19]. A June 12, 2005 experiments caused 7 mail flows per minute, flooding the mailbox[16]. This demonstrates alerting techniques must be tuned to the environment. Outbound traffic might potentially be a cause. This details the warning. Other than TCP and UDP, other protocols should notify. TCP contains 6(decimal)

and UDP 17 in the IP header (decimal)[20]. Other values might indicate unknown assaults and evade firewall rules[20]. Accepted outbound traffic requires training the alarm system with legitimate traffic patterns. When an attacker finds a new approach to exploit unrecognized vulnerabilities, an alarm is missed. Only recognizing patterns isn't enough[20].

2.1.3 Deception Technology

Deception technology is a cybersecurity strategy that involves deploying deceptive elements within a network to misdirect and confuse attackers. It aims to detect, divert, and delay malicious actors, as well as gather intelligence about their techniques, tactics, and motives. Here is a conceptual review of deception technology in cybersecurity:

Purpose: The main purpose of deception technology is to enhance the overall security posture of an organization by deceiving attackers and diverting their attention away from critical assets. It provides an additional layer of defense by actively detecting and engaging with potential threats, reducing the risk of successful attacks.

Deceptive elements: Deception technology employs various decoys, bait, and lures to create an illusion of vulnerability or valuable assets within a network. These elements can be in the form of fake devices, servers, files, or credentials that appear authentic to attackers. They are strategically placed to draw the attention of attackers away from actual assets.

Deception deployment: Deception elements can be deployed across different network layers, such as endpoints, servers, applications, and the network infrastructure itself. They can be either distributed or concentrated in specific areas based on the organization's risk profile and security needs. Proper integration and configuration are necessary to ensure seamless operation and minimal impact on legitimate users.

Detection and alerting: Deception technology relies on the detection of unauthorized activities within the deceptive environment to trigger alerts and initiate response actions. Actions like accessing decoy files, attempting to exploit fake vulnerabilities, or interacting with deceptive systems can generate alerts, indicating potential threats. These alerts are distinct from traditional intrusion detection systems, as they specifically target interaction with deceptive elements.

Incident response: Deception technology plays a significant role in incident response by providing early detection and real-time insights into attacker behavior and intentions. The information gathered from attacker interactions with deceptive elements can be used to develop targeted response strategies, strengthen security measures, and enhance incident handling capabilities.

Integration with other security solutions: To maximize the effectiveness of deception technology, it is essential to integrate it with other security solutions within an organization's cybersecurity framework. Integrating with threat intelligence platforms, security information and event management (SIEM) systems, and forensic tools can enhance the overall threat detection, analysis, and response capabilities.

Challenges and considerations: Deception technology is not without challenges. False positives, where legitimate users accidentally trigger alerts, can be a concern. Organizations need to carefully configure and adapt deception elements to minimize false positives. Regular maintenance and updating of deception assets are necessary to keep them realistic and avoid detection by sophisticated attackers.

Ethical considerations: The deployment of deception technology raises ethical considerations, such as potential entrapment and the risk of targeting innocent individuals

or organizations. It is crucial to establish clear guidelines and adhere to legal and ethical boundaries to ensure the responsible and ethical use of deception technology. In summary, deception technology is a powerful cybersecurity approach that helps organizations proactively defend against attackers by diverting and misleading them. Through the deployment of deceptive elements, organizations can detect threats early, gain valuable insights, and enhance their incident response capabilities. However, proper planning, integration, and ethical considerations are imperative to ensure the successful and responsible implementation of deception technology.

2.1.4 Honeypot System

A honeypot is a computer security system designed to attract and trap intruders. The trapped intruder is then interrogated to determine their motivation and capabilities. If the intruder is not hostile, the honeypot may be used to gather intelligence about their tactics and Techniques. The main purpose of an information security policy is to ensure that services are safe, secure, authentic, available, and accessible[38]. Attacks rely on programs that search a network looking for vulnerabilities, so the honeypot's distinctiveness lies in the fact that it openly displays itself as a vulnerable system likely to attract the attention of hackers[38]. The basic objective of honeypots is to fool the attacker into thinking he can gain control of a genuine operational computer, allowing the admin to study the methods of exposing the attackers, protect against fresh threats, and provide them additional time to react[38]. Honeypots are extremely adaptable and come in a variety of shapes and sizes. Most works define honeypots in 2 directions: the first classifies them based on the interactions they enable, and the second classifies them based on their utility[38].

A honeypot is a security resource whose value is determined by its ability to be explored, exploited, or hacked[38]. This implies that whatever we identify as a honeypot, we anticipate and intend for the system to be examined, attacked, and potentially exploited[38]. Honeypot is primarily a detection and reaction tool, with minimal utility in preventive[38]. Honeypots do not block specific intrusions or the transmission of viruses or worms. Instead, they gather data and detect attack trends[38]. After that, defenders can respond to this evidence by constructing stronger defenses and countermeasures against future security threats[38]. A honeypot is a tool used to gather evidence or information and to learn as much as possible about attack patterns, hacker purposes and motives, and widely utilized programs launched by them[38]. More about the hacker's abilities can also be learned, particularly their technical understanding, based on the information we have acquired[38].

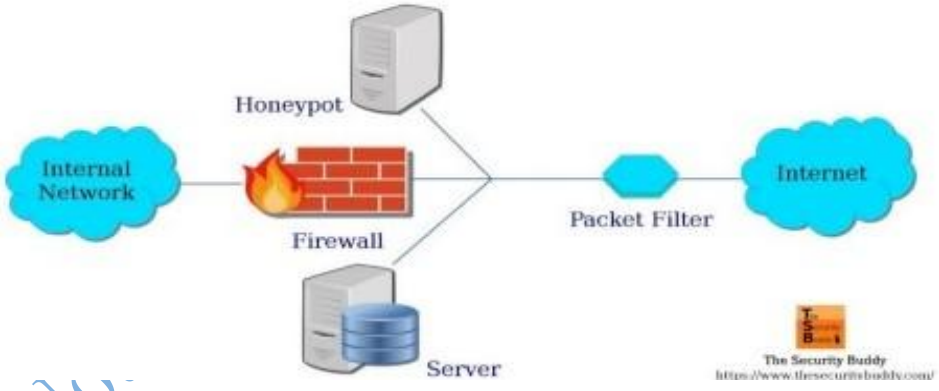


Figure 2.3 Honeypot for Improving Security[38]

Honeypots may capture intruders on the network and deflect them from the production system[39]. Honeypot operators need competence in security, systems, and networks. Honeypots may be useful information-gathering tools[39]. Honeypots may become invading equipment in the wrong, unskilled hands. Honeypots seldom improve network security. Instead, they lure invaders. Figure 2.1 depicts honeypots as useless security instruments.

Nobody should send it[39]. Honeypot activity or traffic might be seen as an intrusion, illegal access, or probing effort. If the honeypot begins outbound connections, it's been hijacked. Shapes and sizes of honeypots vary. Low-interaction and high-interaction honeypots[39].

Honeypots have grown increasingly important in Internet security[40]. Hackers could quickly tell whether a server contained honeypots. Scholars are focusing on techniques to automatically detect a honeypot server to mitigate this threat. Researchers must enhance honeypots' internal mechanism and outward interface[40]. The comparison study of different methods by different authors of honeypot system can be found in Appendix III.

Purpose: The primary purpose of a honeypot system is to act as a defensive mechanism for identifying, studying, and mitigating potential threats. By luring attackers to the honeypot, organizations can gain valuable insights into their tactics, tools, and motives.

Types of honeypots: There are several types of honeypots, including low-interaction honeypots, high-interaction honeypots, and hybrid honeypots. Low-interaction honeypots simulate only a limited range of services and are less resource-intensive. High-interaction honeypots provide a complete operating system and simulate real services, thus offering more detailed analysis.

Deployment: Honeypots can be deployed in various ways, such as on dedicated hardware, virtual machines, or as part of an existing network. The deployment method should protect the integrity of the production network and ensure the honeypot remains isolated to prevent real systems from being compromised.

Data collection: Honeypots capture information about attacker activities, including their attack methods, tools, and exploit techniques. They record network traffic, commands

executed, and interactions with the honeypot, providing valuable insights that can help in improving security measures.

Risks and challenges: While honeypots offer numerous benefits, organizations need to be aware of the risks and challenges associated with their deployment. Attackers could potentially use the honeypot to launch attacks on other systems or gain unauthorized access if not properly isolated. False positives, where legitimate users accidentally interact with the honeypot, can also be a concern.

Legal and ethical considerations: The use of honeypots raises legal and ethical considerations. Organizations need to ensure that they comply with relevant laws and regulations and consider the impact on privacy. Proper consent and disclosure are necessary when deploying honeypots in certain jurisdictions.

Incident response: Honeypots play a vital role in incident response by providing early warning signals and helping in proactive defense. By analyzing the attacker's behavior and tactics, organizations can develop more effective strategies to detect, prevent, and respond to future attacks.

Integration with other security systems: Honeypots can be integrated with other security systems, such as intrusion detection systems (IDS) or security information and event management (SIEM) platforms. This integration helps in correlating and analyzing the collected honeypot data with other security events, enhancing overall threat intelligence. In conclusion, honeypot systems are an effective tool for gathering intelligence on attackers and enhancing an organization's overall cybersecurity posture. However, careful planning, implementation, and consideration of legal and ethical considerations are crucial to ensure their successful deployment.

2.1.4.1 Using A Honeypot to Identify Unknown Attacks

The Honeynet Project that was carried out by Roo explained how honeypot system can be used to identify unknown attacks that might try to attack the sensitive information of the system and display the identities of such an attack[21]. It satisfies all of the aforementioned criteria and detects previously undiscovered attacks. Roo monitors network traffic using the Honeywall and one or more decoys. The Honeynet and Honeywall link the decoy devices to a production network[22]. An operator must record configuration variables before constructing a Honeypot[22]. Changing variables, such as an ISP's changing IP, must be logged. This allows for setup repetition and analytic comparison[22]. A hacked Honeypot threatens other network devices. These might be on the production network or online[22]. Allowing an attacker to infiltrate production equipment might disclose sensitive data. Denial-of-service attacks conducted from a Honeypot may be illegal. Before connecting the Honeypot to public networks, test cases must be built and run[22]. Test scenarios must cover all key features and guarantee the Honeypot doesn't cause greater danger[22].

2.1.4.2 Known Attacks Versus Unknown Attacks

Roo detects suspicious connections using multiple components. Snort is crucial to pattern analysis and identification of attacks. Payload, non-payload, and post-detection rules define Snort patterns. Non-payload rules focus on protocol headers[23]. Post-detection rules take action. Every Honeypot connection includes protocol, IP addresses, ports, packet count, etc. A GUI shows each flow and provides tcpdump1 downloads[23]. Snort lists known attacks[23]. Unknown attacks have a generic alert (e.g., "SHELLCODE x86 inc ebx NOOP"), "unknown signature," or no warning. The shellcode notification isn't a known exploit since it doesn't target a vulnerability. "Noop-sliding" is shellcode technique[23].

2.1.3 Verifying New Attacks

Honeywall doesn't monitor decoy processes, making it difficult to verify attack success and without knowing what each link did, an attack's success cannot be determined. Analyzing decoy answers isn't enough[24]. The attacked system may provide false answers denying the assault[20]. Only replaying an assault can establish its success[24]. A second Honeypot matching the original's prerequisites and variables is required to replicate an assault. Using the original configuration sheet, the operating system is reinstalled[24]. Trial system including monitoring tools and debugger will also be configured to provides access for monitoring the windows[25]. Windows monitors the registry, files, processes, and ports. Function calls and library access are shown[25]. A successful connection produces another[25]. An attacker may use the new connection to download a binary that compromises the machine. A new Snort rule may detect the attack with the verification. Patch production machines and make Honeypot test a bugfix or remedy[25].

2.2 Methodological Review

2.2.1 Cyber Security

A mythological review in cybersecurity refers to an exploration of common myths, misconceptions, and false beliefs that exist within the field. It involves examining widely held beliefs or popular narratives related to cybersecurity practices, technologies, or threats and critically evaluating their accuracy and validity. Here are some examples of areas where mythological reviews in cybersecurity can be conducted:

Password strength: Many people believe that creating complex passwords with a combination of uppercase, lowercase, numbers, and special characters is sufficient to protect their accounts. A mythological review would assess the actual effectiveness of

password complexity and explore alternative methods, such as password managers or multi-factor authentication.

Antivirus software: There is a common belief that having antivirus software installed guarantees protection against all types of malwares. A mythological review would analyze the effectiveness of antivirus software in detecting and preventing sophisticated threats, explore its limitations, and discuss additional security measures that should be implemented.

Phishing awareness: It is often assumed that employees are well-educated about phishing attacks and can easily identify suspicious emails. A mythological review would assess the level of phishing awareness within organizations, identify common misconceptions, and propose strategies to improve employee education and training.

Patching and updates: Some individuals believe that regularly updating their software or operating systems is unnecessary or could cause issues. A mythological review would evaluate the importance of timely patching, consider the risks associated with delayed updates, and recommend best practices for managing software updates.

The role of cybersecurity tools: Many people assume that implementing cybersecurity tools, such as firewalls or intrusion detection systems, provides complete protection against all threats. A mythological review would examine the capabilities and limitations of these tools, discuss their appropriate usage, and highlight the need for a layered defense strategy.

Zero-day vulnerabilities: There is often a perception that zero-day vulnerabilities are the most significant threats, and they cannot be mitigated effectively. A mythological review

would explore the severity and prevalence of zero-day vulnerabilities, assess the effectiveness of various countermeasures, and discuss risk management approaches for handling such vulnerabilities.

Attribution in cyberattacks: The belief that accurately attributing cyberattacks to specific individuals, groups, or nation-states is straightforward and conclusive can be misleading. A mythological review would analyze the complexities and challenges associated with attribution, debunk common misconceptions, and explore different attribution techniques and their limitations. By conducting a mythological review in cybersecurity, the field can gain a more accurate understanding of various cybersecurity practices, technologies, and threats. This helps dispel false beliefs, correct misconceptions, and inform decision-making processes, ultimately leading to more effective and informed cybersecurity strategies and practices.

The two most important security behaviours that any company is concerned with on a daily basis are privacy and data security[26]. Today's predominantly digital or cyber-specific world, where all data is housed, typically surrounds measures. Cybercriminals exploit social networking sites to steal personal information from users[26].

The connected electronic information network has become an integral part of our daily lives[24]. All types of organizations, such as medical, financial, and education institutions, use this network to operate effectively[26]. They utilize the network by collecting, processing, storing, and sharing vast amounts of digital information[15]. As more digital information is gathered and shared, the protection of this information is becoming even more vital to our national security and economic stability[26].

Cybersecurity protects networked systems and data against abuse or harm[26]. You must protect your identity, your data, and your computer equipment on a personal level. At the corporate level, everyone is accountable for safeguarding the company's name, data, and clients. The safety and wellbeing of the populace, as well as national security, are at risk at the state level. Your identification might have an impact on your life as more time is spent online[27]. Your offline persona is the person who regularly engages with your friends and family at home, school, or job. They are aware of details about you, like your name, age, and residence. You are who you are online, according to the internet. How you represent yourself to others online is your online identity. This online persona ought to only provide a certain amount of information about you[27].

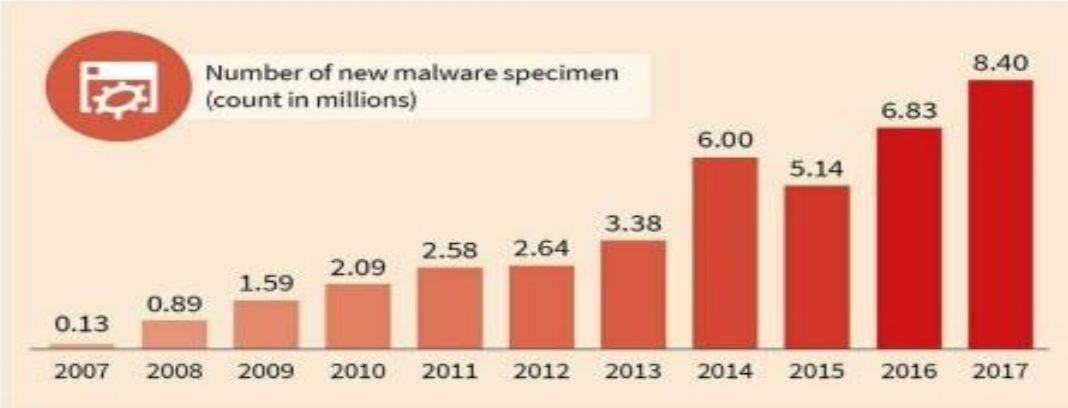


Figure 2.2 New Attacks Evolution[9]

Choose your username or alias carefully. The username shouldn't be personal. Respectful language is needed. This username shouldn't make folks believe you're easy prey[28].

Cyber security protects against cyber-attacks due to vulnerabilities and hazards. Cybersecurity prioritizes data accessibility, integrity, and privacy[28]. Cyber-attacks prohibit cyber security. Cyber-attacks aim to limit services and tamper with data (modification,

destruction, disclosure, sharing). Cyber-attacks fall into 5 categories[28]. Below are fundamental groupings and approaches.

• Service blocking attacks include DoS and DDoS

• Malicious software, including viruses, worms, Trojan horses, keyloggers, adware, spyware, bots, and scareware

• Phishing

• Unwanted Electronic Mail: Spam

• Network Traffic Monitoring: Sniffing

2.2.1.1 Types of Cyber Attack

1. Denial-of-Service (DoS) and Distributed Denial of- Service (DDoS) Attacks

A Denial-of-Service exploit exhausts the system's capacity, preventing it from responding to service requests. A DDoS attack is launched by a host system that has been infected with malicious programs and is managed by an attacker[29]. In this type of cyber-attack, the computer or networks resources are rendered inaccessible to the desired user by interfering with the services of the hosts that are connected to the network. Various sorts of DoS and DDoS attacks include TCP SYN flood attacks, teardrop attacks, smurf attacks, ping-of-death attacks, and botnets[29].

2. Man-in-the-Middle (MitM) Attack

Third-party Man-in-the-Middle attacks include client-server interactions. This required some impersonated client and server to have access to the shared data. This attack steals, transmits, and receives data. MITM employs real-time communications, transactions, and

info-sharing[30].

3. Phishing Attacks

False messages that appear to be from trusted sources are called Phishing. Its primary goal is to obtain sensitive information such as personal information and credentials[30]. It is a type of social engineering or technological deception that uses emails with embedded URLs to send malicious files to our system[30].

4. Drive-by- Download Attack

Drive-by-download attacks distribute viruses and give hackers unauthorized access. Infected machines access a website[30]. They are indeed a prevalent type of cyber-attack used by hackers to propagate viruses and acquire illicit access[30]. This type of attack happens whenever a computer is compromised with malicious programs merely by browsing a website[30].

5. Password Attack

Passwords are more commonly used means of user authentication, and acquiring such credentials is an effective attacking strategy. A credential hack is a process where a user's credentials are stolen or decoded illegally[31]. Finding the password may include checking the user's workstation, assuming, collecting a login database, or monitoring the network connection[31].

6. SQL Injection Attack

An attacker who wishes to do SQL injection may alter a regular SQL query to exploit unsubstantiated weaknesses in a database. Mis-filtered characters can also be used by attackers to change SQL statements[31]. SQL injection exploits alleged database flaws by

modifying a typical SQL query. There are various viable methods for preventing and defending against SQLI attacks if they occur[31].

7. Cross-Site Scripting (XSS) Attack

Cross-Site Scripting is a form of injecting technique in which a malicious script injects a malicious script into a site or application. In other word, Attackers inject XSS into a website's database. The victim's browser runs the infected script inside the response, transmitting the victim's cookies to the attacker's server[32].

8. Eavesdropping Attack

This is known as snooping attacks or spying attacks. Eavesdropping breaches and compromises digital data[32]. Attackers exchange communications over an unsecured network and examine data sent and received. An attacker may use a sniffer to eavesdrop on a computer or server to grab data while it is being transmitted during this type of attack, which is hard to identify since it exhibits no anomalous behaviour during network transmission[32].

9. Birthday Attack

Birthday attacks are a type of cryptographic attack that falls into the brute force attack category. It is based on the birthday problem premise from probability theory[33]. This technique could be used to exploit the sharing of data among more than two parties. Birthday attacks are performed out by employing hash methods to validate the message's authenticity, program, or cryptographic signature[33].

Birthday assaults are a brute-force cryptography attack. Based on probability theory's birthday issue[33]. This approach exploits data exchange between more than two parties.

Birthday attacks authenticate a message's validity, software, or cryptographic signature using hash algorithms[33].

Figure 2.3: Types of Attacks[33]

10. Malware Attack

Malware attacks are a type of cyber threat wherein a harmful program is placed on the victim's machine even without the user knowing or agreeing to it[34]. Viruses, malware, and ransomware are just some of the terms that are currently used when referring to pernicious software. Malicious software propagates itself and executes by itself[34]. Through malware, access to a private network can be gained, disrupted, and personal data can be taken as shown in figure 2.2 or other user information can be accessed, making it possible to illegally earn money[34]. Malware now mostly targets commercial or financial data rather than personal sensitive information. The most frequent types of malwares are:

a. Virus: a harmful program that attaches itself to any system program and replicates and modifies instructions when run. It may spread by accessing files or by launching any software[34].

b. **Worms:** These are viruses that propagate through devices or the internet via email attachments. This might lead to denial-of-service attacks[34].

A malware called Trojan Horses is a type of malicious software that masquerades as a helpful software and does not spread like viruses.

c. Ransomware: A malicious software that encrypts user data and threatens the user until a ransom is paid. Even though the code is basic, it is quite difficult to avoid this attack[34].

d. Spyware: Malware that examines user activities without the user's permission and reports this to the hacker[34].

2.2.1.2 Trends Changing Cyber Security

The following are a few factors with a significant impact on cyber security:

1. Mobile Networks: This allows electronic communication or sending information between two or more persons across the globe. However, security has become a troubling issue for these mobile networks were attacker tries to hijack information through a medium and make sure the information is destroyed or not getting to the destination. The deployment of firewalls and other security tools are becoming inefficient and increasingly permeable as people use smartphones, tablets, PCs, and so on[35].

2. Web servers: Web servers are vulnerable to web application hacks that can steal data or distribute harmful code[35].

3. Cloud computing and its services: Businesses of all sizes are gradually adopting cloud-based services these days. In other words, the entire planet is gradually moving toward the CLOUD[35].

4. APTs and targeted attacks: There is a completely new type of cybercrime ware known as APT (Advanced Persistent Threat)[35].

5. Code encryption: Communication (or information) is encrypted so that it cannot be interpreted by eavesdroppers or attackers[35].

6. IPv6: 'IPv6' stands for 'Internet Protocol 6'. It will replace IPv4, which has served as the foundation of the networks and the Internet in general[35].

2.2.1.3 Cyber Security Techniques

Cyber-attacks on the internet can expand by using new approaches. To exploit new technological flaws, cybercriminals would constantly modify existing malware fingerprints[37]. In other cases, they look for unique properties of new technologies to uncover flaws in virus insertion[37]. Cyber thieves are trying to take benefit of developing Internet technology and millions or billions of active users to gain easy and efficient access to a large number of individuals[37].

The following are the major techniques been adopted in cyber security to provide safety and privacy to the information in the cloud server.

- Access control and password security
- Data authentication
- Malware scanners
- Anti-virus software
- Firewalls

2.2.2 Deception Technology

A methodological review in deception technology focuses on the research methodologies and approaches used to study, analyze, and evaluate deception techniques and technologies in cybersecurity. It involves examining the strengths, limitations, and best practices of different research methods to guide future studies and advancements in the field of deception technology. Here are some key methodological areas in deception technology:

a. Experimental studies: Experimental studies involve controlled settings where researchers manipulate variables to measure the effectiveness of deception techniques. This method allows for evaluating the impact of different deceptive elements, such as bait assets, lures, or honeytokens, on attacker behavior or system defenses. It may involve simulations, virtual environments, or real-world experiments to gather data and draw conclusions.

b. Case studies: Case studies involve in-depth investigations of specific instances or scenarios where deception technology has been deployed. Researchers analyze real-world deployments to gain insights into the implementation challenges, effectiveness, and impacts of deception techniques. Case studies often include qualitative data collection methods, such as interviews, observations, and document analysis.

c. Surveys and questionnaires: Surveys and questionnaires involve collecting data from a large number of participants to understand their perceptions, experiences, or attitudes regarding deception technology. This method provides quantitative data that can be statistically analyzed to identify trends, preferences, or barriers related to the adoption and use of deception techniques. Surveys may cover a range of topics, including user acceptance, usability, or cost-effectiveness.

d. Threat modeling and analysis: Threat modeling and analysis involve the systematic identification, assessment, and categorization of potential threats and their associated risks. Researchers may use threat modeling techniques to study how deception elements can be strategically placed within a system to deceive attackers or disrupt their actions. This method helps in understanding how deception technologies can influence an attacker's decision-making process.

e. Simulation and modeling: Simulation and modeling methods involve building computational models or simulations to study the behavior and interactions of attackers and defenders in different cyber threat scenarios. Researchers can use these models to analyze how deception techniques impact attacker behavior, system defenses, and overall cybersecurity posture. Simulation techniques can also be used to test and validate the effectiveness of newly proposed deception technologies.

f. Ethnographic studies: Ethnographic studies involve observing and immersing oneself in the real-world contexts where deception technology is deployed. Researchers may spend time with users, defenders, or attackers to understand their practices, motivations, and experiences. This method provides rich qualitative data that can help uncover unforeseen challenges, cultural influences, and societal implications related to the use of deception techniques.

g. Evaluation frameworks and metrics: Methodological reviews in deception technology often focus on evaluating the effectiveness and efficiency of deception techniques. Researchers may develop evaluation frameworks and metrics that define criteria for assessing various aspects, such as attacker detection rates, response times, false positive

rates, or the impact on attacker dwell time. These frameworks and metrics help establish standardized approaches for comparing different deception technologies and determining their practical value. A methodological review in deception technology helps researchers and practitioners understand the strengths and limitations of various research approaches, enabling them to design rigorous studies and advance the field. By critically examining existing research methodologies, gaps in knowledge can be identified and strategies can be developed to address those gaps, leading to the improvement and innovation of deception technologies in cyber defense.

2.2.3 Honeypot

A methodological review in honeypots focuses on the research methodologies and approaches used to study, analyze, and evaluate the effectiveness and usage of honeypots in cybersecurity. It involves examining the strengths, limitations, and best practices of different research methods to guide future studies and advancements in the field of honeypot deployment. Here are some key methodological areas in honeypot research:

a. Deployment strategies: Honeypots can be deployed in various ways, such as network-level, host-level, or application-level honeypots. A methodological review looks at the different deployment strategies used in previous research to assess their advantages, limitations, and suitability for specific use cases. It examines topics like decoy network design, honeypot placement, and integration with existing network infrastructure.

b. Data collection and analysis: Honeypots are primarily used to collect data about attacker behavior and techniques. Methodological reviews analyze the data collection techniques employed in honeypot research, including network traffic monitoring, system

event logs, and session recordings. It also examines the various tools and methodologies used for analyzing the collected data to gain insights into attackers' motivations, tactics, and potential vulnerabilities in the targeted systems.

c. Attack simulation: Researchers often simulate attacks against honeypots to understand attacker actions and intentions. Methodological reviews evaluate the different attack simulation techniques used, such as automated attacks, targeted attacks, or social engineering techniques. It examines the benefits and drawbacks of each approach and assesses how well these simulations represent real-world attack scenarios.

d. Evaluation metrics: The effectiveness of honeypots is measured using various evaluation metrics, such as time to detection, attacker engagement, or false positive rates. Methodological reviews examine the different metrics employed by researchers to determine the strengths and limitations of each metric. It also explores the development of standardized evaluation frameworks that enable fair comparisons between different honeypot solutions.

e. Longitudinal studies: Honeypots, when deployed for a significant duration, allow for a longitudinal study of attacker behavior over time. Methodological reviews assess previous studies that have utilized long-term honeypots to understand trends, changes in attack patterns, and evolving attack techniques. It examines the challenges and techniques of maintaining honeypots over extended periods and the analysis of long-term data.

f. Ethical considerations: Honeypot research raises important ethical considerations regarding data privacy, legal implications, and potential harm to attackers or innocent parties. Methodological reviews analyze how researchers have addressed these ethical

concerns and adhered to ethical guidelines in their research. It looks into methodologies that ensure compliance with legal and ethical standards when conducting honeypot experiments.

g. Operational considerations: Honeypots need to be carefully managed and monitored to ensure their effectiveness and avoid misuse. Methodological reviews examine the practical aspects of deploying and maintaining honeypots, including configuring system monitoring, maintaining honeypot authenticity, and handling captured data securely. It also explores methodologies for minimizing the risk of honeypot detection and evasion by attackers. A methodological review in honeypots helps researchers and practitioners understand the strengths and limitations of different research approaches, enabling them to design more effective honeypot deployments and studies. By critically examining existing methodologies, gaps in knowledge can be identified, leading to the development of improved honeypot solutions and more comprehensive evaluations.

2.2.3.1 Review Based on Machine Learning Algorithms

Many network intrusion detection system studies incorporate machine learning in their operation to make it effective[42]. Open-source datasets have helped find and avert hazards. Machine learning and data science combine to produce hybrid systems that employ layered and hierarchical models, discover anomalies, and complement machine learning with knowledge-based approaches. Most of these systems recognize fresh threats while reducing false alarms[42]. Intrusion detection uses SVMs, fuzzy logic, and neural networks.

IDS combines J-Rip, REP trees, and Forest PA with rule-based and decision-tree methods. Input characteristics classify the first two techniques[42]. Unlike REP trees, Forest PA uses both classifiers. CICIDS-2017 yielded 97% accuracy and 95% detection. Research uses hybrid methods. NBC and NB-Tree enhanced recall and accuracy[42]. Using constrained feature sets and regular testing, this strategy combines accuracy and recall for small and big threats. IDS false-positives are manageable. This approach classifies minor groups with 99.05% accuracy[42].

It was discovered that a neural network-based IDS can identify attacks on the network. To identify intrusions, neural network topologies are compared. MLP found offline incursions[42]. Early validation improved the neural network's reasoning abilities. A two-hidden-layer neural network effectively detected logs with 91% accuracy[42]. ANNs can detect intrusions[57]. Choosing a viable neural network framework is difficult[42].

They needed a more accurate, computationally expensive intrusion detection approach based on researcher's method. Data is protected by three tiers[43]. Connection setup, user identity, and destination IP addresses are first-layer packet characteristics. Availability detects probing, U2R, R2L, and DoS. It acquires data, views files, etc. The author created a multi-layered technique[43]. The writers evaluate each tier's data integrity, permissions, and file updates. Nave Bayes, C5 decision tree, and MLP neural network methods were used[39]. Naive Bayes, C5 decision trees, and MLP neural networks with gain ratios enhanced storage and performance[43]. Multiple-layer model proved more accurate than MLP neural networks and Nave Bayes, reducing false alarms. This method identifies invasions[43].

2.2.3.2 Review Based on the Honeypot Techniques

Recent attention on hostile incursions has created a bottleneck which has made network security difficult to operate. All security measures have been agreed to be faulty in one way or the other when comparing their techniques in terms of operation and the level of loopholes through which attackers could penetrate. Most current research uses machine learning to identify honeypot incursions[424].

A machine-learning honeypot is designed to improve IoT device security. An IoT scanner analyses the internet for potentially harmful interactions and teaches the honeypot how to optimize each response[44]. This attack classifies using anomaly and honeypot data[44]. The authors recommend automating spam classification using SVMs and social honeypots[44].

Researchers recommend a linkage-protected honeypot[44]. Protocols link honeypots to defense system parts which depends on the honeypot's condition[44]. The honeypot analyses defensive flows. If the honeypot is disrupted, it tends to offer a cloud-based AV installations and scans[44]. The researcher's novel strategy used retroactive software identification. Realize Cloud AV[44].

Another researcher proposed providing virtual instance binary program to analysis engines[45]. Each system call is examined in real-time[45]. Honeypots detect bugs and viruses. A honeypot captures, registers, and analyses attackers' activity[45]. The authors classified things using cluster-based approaches[45]. This approach is self-taught. A researcher employed machine learning to construct an IoT honeypot model. Honeypots improve model management[45]. A machine learning model for Myspace, Facebook, and Twitter was created and tested[45].

An author created a honeypot model from Feng's link protection mechanism. SNMP increases communication and management in this paradigm[45]. Honeypots monitor defenses to stop or enable assaults. Assault was predicted using machine learning, machine learning and honeypot algorithms are proposed[45]. Machine-learning and honeypot algorithms prevent intrusions[45].

Honeypot latency[45]. In certain virtual networks, honeypots like honeyd cause connection delays of one to 10 milliseconds[45]. Neyman-Pearson theory increased their detection rate. Network monitoring or virtual environments may reveal honeypots[45]. In the course of reviewing some articles on honeypot system, honeypot network-level activities and services as identified, while another author created honeypot detection using Linux and virtual machine files[45]. The comparative study of different approach by various authors can be found in Appendix VI.

In recent years, more assaults have targeted these sites. 20% of cyberattacks in 2020 will be cloud-based[46]. Therefore, it's necessary to create a system that can pose as a decoy to the attacker while monitoring the attack and gathering actionable data to avoid repeat attempts. Honeypots are virtual systems meant to replicate the behaviour of actual assets[46]. Honeypots have several flaws. The system was meant to be attacked thus assaults are expected[47]. Once the honeypot is penetrated, it may launch subsequent assaults. These assaults might target an internal or external system. Honeypots are risky. Legal culpability results. If your honeypot attacks another firm, you might be sued. Honeypots influence risk[47].

Cloud infrastructure is being used more and more to solve complicated challenges. No system is invincible, therefore cyber-attacks continually threaten cloud solutions[47]. As

known, cloud systems have built-in defensive measures, but a honeypot adds an additional layer of protection while revealing attack vectors. To solve the issue, create a hybrid honeynet system that combines the best of several honeypots[47].

2.2.4 Security Vulnerabilities

Security vulnerabilities are any kind of software or hardware defect. After gaining knowledge of a vulnerability, malicious users attempt to exploit it[48]. An *exploit* is the term used to describe a program written to take advantage of a known vulnerability[48]. The act of using an exploit against a vulnerability is referred to as an attack. The goal of the attack is to gain access to a system, the data it hosts or to a specific resource[48].

2.2.4.1 Software Vulnerabilities

Software vulnerabilities are usually introduced by errors in the operating system or application code. Despite all the effort companies put into finding and patching software vulnerabilities, it is common for new vulnerabilities to surface[48]. Microsoft, Apple, and other operating system producers release patches and updates almost every day. Application updates are also common. Applications such as web browsers, mobile apps and web servers are often updated by the companies or organizations responsible for them[48].

Cisco IOS has SYNful Knock vulnerability in 2015. This weakness lets attackers control Cisco 1841, 2811, and 3825 routers. Attackers monitored all network activity and infected new machines[49]. The routers' updated IOS caused this vulnerability. Verify the downloaded IOS image's integrity and limit physical access to the device to avoid this[49].

The goal of software updates is to stay current and avoid exploitation of vulnerabilities[49]. While some companies have penetration testing teams dedicated to search, find and patch software vulnerabilities before they can get exploited, third-party security researchers also specialize in finding vulnerabilities in software[49].

Google's Project Zero is a great example of such practice. After discovering a number of vulnerabilities in various software used by end-users, Google formed a permanent team dedicated to finding software vulnerabilities[49].

2.2.4.2 Hardware Vulnerabilities

Design errors generate hardware vulnerabilities. RAM memory has close-packed capacitors. Due of proximity, constant changes to one capacitor might affect others. Rowhammer was based on this design flaw of the existing system. The Rowhammer attack retrieves data from neighbouring protected address memory cells by repeatedly rewriting memory at the same places[60].

Model-specific hardware vulnerabilities are seldom exploited randomly. Standard malware protection and physical security are adequate for the ordinary user[60].

2.2.5 Categories of Security Vulnerability

The bulk of software security issues are:

a. Buffer Overflow: This happens when data exceeds a buffer's bounds, buffers are program-assigned memory areas within a computer memory location. By altering data beyond a buffer's limits, the application accesses other processes' memory which may cause system failure, data leak, or privilege elevation[61].

Programs are designed to make use of non-validated input for its execution, the data may include malicious code to pressure the application to behave badly. Consider image-processing software. Malicious users may produce picture files with wrong dimensions. Malicious dimensions may lead applications to produce buffers with erroneous sizes[61].

b. Race Conditions This occurs when an event's output is ordered or timed. When events aren't arranged or scheduled properly, a racing scenario becomes risky[61].

c. Weak Security Practices - Authentication, authorization, and encryption can protect systems and sensitive data. Developers that build their own security algorithms risk creating weaknesses. Developers should utilize proven, certified security libraries[61].

d. Access-Control Problems – Access control regulates who does what, from controlling physical equipment access to prescribing who may view or alter a file. Misusing access limitations poses security issues[61]. If an attacker has physical access to target equipment, they can bypass most security measures. No matter how you set a file's permissions, the operating system can't stop someone from reading its contents from the disc. Physical access must be limited and encryption utilized to avoid data theft and corruption[61].

2.2.12 Types of Malwares

Malware steals data, bypasses access controls, damages, or undermines a system. Malwares:

1. Spyware tracks and spies on the user. Spyware records behaviour, keystrokes, and data. Spyware modifies security settings to hide. Spyware integrates with legitimate software or Trojans[62].

2. Adware automatically delivers ads. Software contains adware. Adware is typically bundled with spyware[62].

3. Bot - Autonomous internet malware. Botnets are an increasing threat among information system developers. It has been discovered that several computers have waitbots to contend with[62].

4. Ransomware holds a machine or its data hostage until a payment is paid. Ransomware encrypts data using a key. Some ransomware variations lock machines via system weaknesses. Ransomware spreads via downloads and software vulnerabilities[62].

5. Scareware is a sort of virus that induces dread in the victim. Scareware pops up OS chat windows. These windows offer fake messages saying the system is in danger or needs to restart. If the user runs the program, his or her PC will be infected[62].

6. Rootkit modifies the OS to create a backdoor. Backdoors provide remote computer access. Rootkits use software weaknesses to modify system files and raise privileges. Rootkits modify forensics and monitoring program, making them hard to spot. Reinstalling rootkit-infected systems is common[62].

7. virus - A virus is harmful code attached to executable files, frequently genuine program. Most infections are user-activated and timed. Viruses might be harmless, showing images, or dangerous, changing or erasing data. Mutating viruses evade detection. USB devices, CDs, network sharing, and email distribute viruses[62].

8. Trojan horse - Malware that performs destructive actions as a desirable action. It abuses the user's privileges. Image files, music files, and games often include Trojans. Trojan horses aren't viruses since they infect non-executable files[62].

9. Worms - Worms proliferate by exploiting network weaknesses. It slows networks once it has a means of penetrating the connection. Unlike viruses, worms may operate independently[62]. After the first infection, users aren't needed. After infecting a host, the worm spreads fast over the network. Worms are alike. All have a weakness, a mechanism to spread and a payload worms cause many Internet assaults[62].

10. Man-In-The-Middle (MitM) – MitM lets an attacker operate a device without user awareness. An attacker may intercept and steal user data with such access. MitM attacks take money. MitM malware, methods abound[62].

11. Man-in-the-Mobile (MitMo) – MitMo is a mobile assault similar to man-in-the-middle. Infected mobile devices may send user data to attackers. MitMo vulnerability ZeuS steals 2-step verification. Texting[62].

2.2.7 Vulnerability and Penetration Tests

System vulnerabilities must be found to effectively detect cybersecurity concerns in the organization's network. Vulnerability lets a network or process intervene[31]. Security or vulnerability evaluations discover network flaws that might lead to a breach. Vulnerability study shows the consequences of vulnerabilities by infiltrating network systems[62]. This distinguishes it from penetration testing[62]. Internationally qualified penetration testing professionals uncover logic mistakes and system vulnerabilities. Penetration tests detect system vulnerabilities without causing damage[62]. Types of penetration or vulnerability tests used to analyze organization systems' cybersecurity includes[62].

- Web Application Tests
- Network Tests
- Mobile Tests

- Client-Side Tests
- Exclusion Tests
- Wireless Network Penetration Tests
- Database Tests
- Social Engineering Tests

2.2.8 Vulnerability Exploitation

Infiltration sometimes involves exploiting weaknesses of the vulnerable system which make attackers to search computers for information for frequent exploit[63].

Step 1: Gather information about the target system. Port scanners or social engineering may be utilized. Learn as much as you can about the target computer[63].

Step 2: One of the key pieces of information gathered in step 1 might be the operating system, its version, and a list of services running on it[63].

Step 3: Once the target's operating system and version are identified, the attacker searches for any known vulnerabilities related to that version of OS or other OS services[63].

Step 4: When a vulnerability is discovered, the attacker employs a previously designed exploit. If no vulnerability exists, the attacker may create one[63].

2.2.9 Symptoms of Malwares

Common malware symptoms include:

- There is an increase in CPU usage.
- There is a decrease in computer speed.
- The computer freezes often.
- Web speed slows.

• Unexpected network troubles.
• Files are modified.
• Files are deleted.
• There is a presence of unknown files, program, or desktop icons.
• There are unknown processes running.
• Programs restart or shut off.
• Unauthorized email sending.

2.2.10 Social Engineering

Social engineering is a kind of access attack that attempts to persuade individuals[64]. People's compassion and flaws are used by social engineers. An attacker may call an authorized employee to report an urgent network problem. The attacker may use the employee's ego, authority, or greed[64].

1. Pretexting is a kind of social engineering attack in which an attacker calls someone and promises to get sensitive information. An attacker claims to need personal or financial information to authenticate the recipient's identity[64].

2. Tailgating is when an attacker follows a victim into a secure location.

3. Something for Nothing (Quid pro quo) - An attacker requests personal information in exchange for a gift[64].

2.2.11 WiFi Password Cracking

Wi-Fi password cracking reveals a network's password. Some password-cracking techniques:

1. Social Engineering — It is term used for a broad range of malicious activities accomplished through human interaction.

2. Brute-Force Attacks — The attacker attempts several potential passwords. If the password is 4 digits, the attacker must attempt all 10000 possibilities. Word-list files are used in brute-force assaults. This file contains dictionary terms[64]. A software attempts frequent word combinations. Complex passwords are harder to guess with brute-force attempts. Examples are: Ophcrack, L0phtCrack, THC Hydra, RainbowCrack, and Medusa[64].

3. Network Sniffing – By listening to and recording network packets, an attacker may learn an unencrypted password (in plain text). Attackers can crack encrypted passwords[64].

4. Phishing- A malicious actor sends a false email from a trusted source. The email wants the recipient to download malware or provide personal information. Phishing is when a bogus email from a retailer urges a customer to click a link[61]. The link might lead to phishing or malware[64].

Spear phishing targets specific individuals. Spear phishing emails are personalized to a single individual, unlike phishing emails[64]. Before transmitting, the attacker investigates the target's interests. An attacker learns the victim likes automobiles and wants a certain model. The attacker joins the target's vehicle forum, forges a sale offer, and emails the victim. The email included automobile photos. Malware is installed when the victim hits the link[64].

2.2.12 Advanced Persistent Threats

One way in which infiltration is achieved is through Advanced Persistent Threats (APTs). They consist of a multi-phase, long term, stealthy and advanced operation against a specific

target. Due to its complexity and skill level required, an APT is usually well funded. An APT targets organizations or nations for business or political reasons[65].

Usually related to network-based espionage, APT's purpose is to deploy customized malware on one or multiple of the target's systems and remain undetected. With multiple phases of operation and several customized types of malwares that affect different devices and perform specific functions, an individual attacker often lacks the skill-set, resources or persistence to carry out APTs[65].

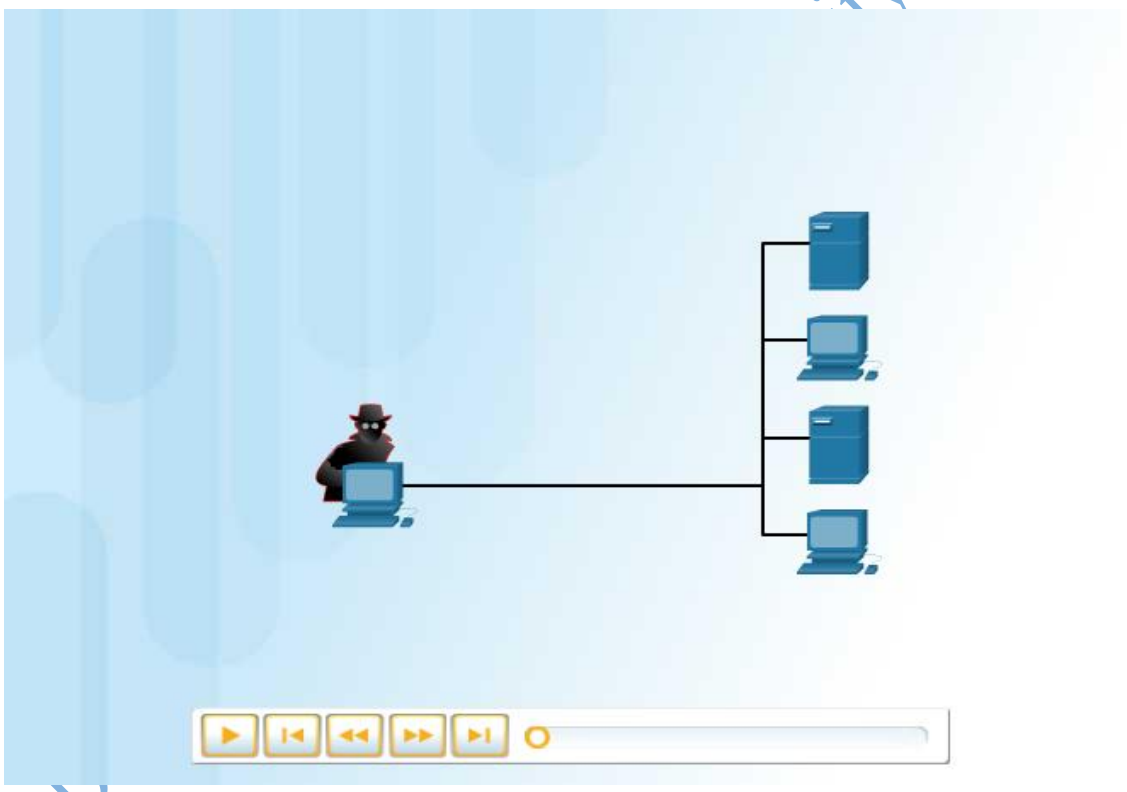


Figure 2.4 Advanced Persistent Threat[65]

2.2.13 DoS Attack

1. Denial-of-Service - DoS attacks are network assaults. DoS attacks affect people, devices, and apps. Two types of DoS attacks exist:

2. Overwhelming Quantity of Traffic - A network, host, or application is inundated with too much data. This causes transmission delays, device or service failure[65].

3. Maliciously Formed Packets - This occurs when a host or application receives a maliciously formatted packet that it cannot process. For instance, an attacker may transfer packets that include defects that cannot be recognized by the application, or packets that have been poorly structured. This causes the receiving equipment to operate at a reduced speed or crash[65]. DoS attacks are seen as a big threat since they may quickly disrupt communication and result in huge financial losses. These attacks are generally straightforward to execute, even for untrained attackers[65].

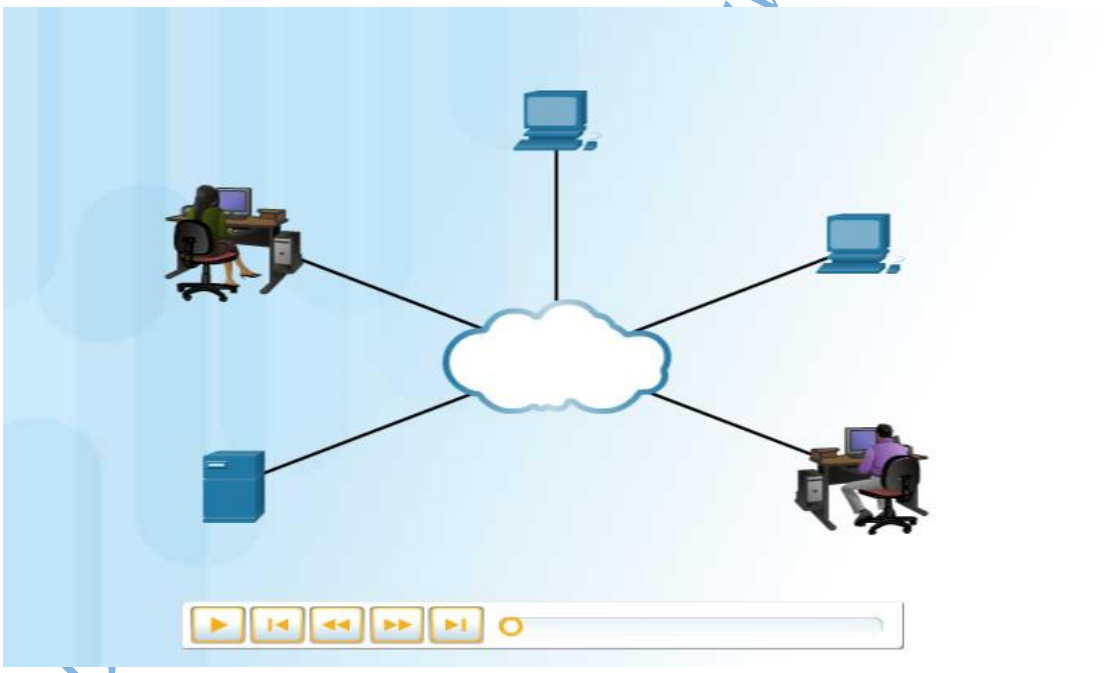


Figure 2.5: DoS[65]

2.2.14 DDoS Attacks

Distributed Denial of Service Attacks (DDoS) is comparable to DoS but comes from several, coordinated sources. Example DDoS Attacker creates botnet. Zombie infected

hosts[65]. Zombies search and infect new victims, multiplying. Hacker instructs DDoS handlers when ready[65].

2.2.15 SEO Poisoning

Google ranks sites and returns relevant results based on user queries. Websites may appear higher or lower in search results based on relevancy. SEO boosts a website's SERP[65]. Criminals may use SEO to enhance a detrimental website's rating[65].

SEO poisoning drives visitors to malicious or phishing sites. Popular search queries improve a rogue site's position[65].

2.2.16 Impact Reduction

Most successful companies recognize fundamental security dangers and seek to prevent them, but no security policy is 100% effective. Large awards must be controlled by companies and organisations[66]. When a security risk is detected, several experts propose these steps:

• Inform others: Staff should be informed and tasked internally. Customers should be notified directly and formally. Communication creates transparency[66].

• Accept the blame if the company errs.

• Specify. And why? The company should cover identity theft protection.

• Identify the cause and facilitator; Recruit forensics experts if required.

• Prevent future intrusions using forensics.

• Ensure all systems have no backdoors BECAUSE backdoors help future attacks

• Educate employees, partners, and customers on breach prevention.

Figure 2.6 Impact Reduction[66]

2.2.17 Types of Data in Cyber Security

Data is anything about individual or organization that can be processed either through electronic or manual means to get meaningful information that can help an organization for decision making. These details of data tend to identify where the data belong[67]. This data includes online photos and communications. Name, SSN, date and place of birth, and mother's maiden name can be used to identify an individual or individuals. Online identifiers include medical, educational, financial, and job information[67].

1. Medical Records

Every time a person or individual goes to the doctor's office, more information is added to his / her electronic health records (EHRs)[67]. The prescription from your family doctor becomes part of your EHR. An individual EHR includes the physical health, mental health, and other personal information that may not be medically-related. For example, if a child

had counseling when there were major changes in the family, this will be somewhere in the child's medical records[67]. Besides one's medical history and personal information, the EHR may also include information about the individual's family[67].

Medical devices, such as fitness bands, use the cloud platform to enable wireless transfer, storage and display of clinical data like heart rates, blood pressures and blood sugars. These devices can generate an enormous amount of clinical data that could become part of your medical records[67].

Fitness bands employ the cloud to wirelessly transport, store, and display clinical data including heart rates, blood pressures, and blood sugars. These devices can produce a lot of clinical data[67].

2. Educational Records

Grades, test scores, attendance, courses completed, accolades, and disciplinary records are in education record. This document may include contact details, health and immunization data, and IEPs[67].

3. Employment and Financial Records

A person's financial record may include information about his / her income and expenditures. Tax records could include paycheck stubs, credit card statements, credit rating and other banking information. Employment information can include past employment and performance data[67].

2.2.18 Data Location

The location of data of an individual or organization is solely depended on the people. It's all about who have access to the privacy of the data. Every nation has various privacy and data regulations that is being managed by people[68]. Where's person's data? the doctor records the chat in the medical chart through electronic patient medical record. The insurance company may get this information to guarantee proper invoicing and quality. Now, the insurance company has part of one's medical record[68].

Store "loyalty cards" may save customers' money[68]. The shop creates a profile customer's purchases for its own benefit. The buyer's profile reveals they buy a specific toothpaste brand and taste[68]. The shop utilizes the buyer's information to send marketing partner offers. Using a loyalty card, the retailer and marketing partner may track a customer's spending[68].

Does someone's know who may have access to online photos if it is shared them with friends? With the devices have the photos in the memory location. Friends may have downloaded such photos[68]. Strangers may have copies of public photos and might download or snap screenshots. Since the photos were put online, they're stored on servers throughout the globe which make the photographs are everywhere[68].

1. Computing Device

Computers don't merely store data nut also process it to become information. These gadgets are data gateways and they produce personal information. Unless one has selected paper statements for all the accounts, such person utilizes computer[68]. For a digital copy of most current credit card statement, visit the issuer's website. To pay one's credit card bill online, visit bank's website to transfer payments. The gadgets may also create information

about someone[68]. With all this information that are available online, it makes one's personal data has become profitable to hackers.

2.2.19 Types of Organizational Data

1. Traditional Data

Corporate data are personal, intellectual, and financial data which includes: application forms, wages, offer letters, and employee agreements[69]. Patents, trademarks, and new product plans are competitive advantages[69].

This intellectual property can be considered a trade secret; losing this information can be disastrous for the future of the company. The financial data, such as income statements, balance sheets, and cash flow statements of a company gives insight into the health of the company[69].

2. Internet of Things and Big Data

With the emergence of the Internet of Things (IoT), there is a lot more data to manage and secure. IoT includes sensors and devices[70]. Cloud and virtualization lead to exponential data expansion. This data has created a new area of interest in technology and business called "Big Data". "Big Data" is a new tech and business trend. With IoT and corporate data's speed, volume, and variety, confidentiality, integrity, and availability are crucial[70].

2.2.20 Cyber Criminals Needs

1. They want your Money

Criminals desire anything valuable such as Valuable internet credentials. Thieves may access accounts using these credentials that is vulnerable[71]. Cybercriminals may be interested in frequent flyer points. Reconsider that Cybercriminals bought free tickets and

upgrades after hacking 10,000 American Airlines and United accounts. Even if airlines refunded the miles, this shows the worth of login passwords. Criminals may also exploit relationships[71]. They might access one's internet accounts and reputation to deceive the into transferring money. The offender might send messages saying your relatives or friends lost their wallets overseas and need money[71]. Criminals are incredibly creative when attempting to steal money. They take your money, identity, and life.

2. They want your Identity

Criminals also seek long-term gain by taking one's identity[71]. Medical identity theft is increasing in tandem with medical expenses. Identity thieves may steal one's medical insurance and use its advantages for themselves[71].

Cybercriminals exploit yearly tax filing systems, which differ by nation. Americans must submit taxes by April 15 each year. The IRS checks the tax return against employer information in July[71]. A tax return may be faked and the refund stolen. When IRS rejects returns, genuine filers notice. They may create credit card accounts and rack up bills using your stolen identity. This will hurt customer's credit and make getting loans harder however, Personal credentials may unlock business and government data[71].

2.2.21 Confidentiality, Integrity and Availability

The CIA trinity guides an organization's information security[110]. Data confidentiality is ensured through authentication encryption. Integrity ensures reliable information. Availability guarantees authorized access to information[72].

1. Confidentiality

Privacy is synonymous with confidentiality which allows only authorized staff should have access to company data, according to company policy. Data may be segregated by security

or sensitivity according tp the level of importance[72]. A Java programmer shouldn't have access to employee data. Employees should be trained on how to secure sensitive information to protect themselves and the firm[72]. Data encryption, login ID and password, two-factor authentication, and reducing sensitive information exposure[72].

2. Integrity

Integrity is data correctness, consistency, and trustworthiness throughout its lifecycle. Unauthorized parties must not modify data during transit[72]. User authorization and access control prevent illegal access. Version control prevents authorized users' inadvertent modifications. Checksum hashing may be used to verify data integrity during transport[72].

A checksum confirms the integrity of local network or Internet-transmitted files or character strings. Hash function checksums[72]. Checksums include MD5, SHA-1, SHA-256, SHA-512. Hash functions convert data to fixed-length values using arithmetic[72]. Comparing hash values. Hashed data can't be decrypted. Hashed value can't retrieve a lost password. Password-reset[72]. After downloading a file, compare its hash value to the source's. Comparing hash values ensures the file wasn't damaged during transit[72].

3. Availability

Maintenance, repairs, updated OS and software, and backups ensure network and data availability[73]. Prepare for disasters, natural or manmade. Firewalls protect against DoS attacks (DoS). Overloading resources creates DoS[73].

2.2.22 Consequences of Security Breach

To protect an organization from every possible, cyberattack is not feasible, for a few reasons. Setup and maintenance of a secure network may be costly[74]. Attackers will

continue to attack networks. Advanced cyber-attacks will succeed. Security staff must react fast to minimize data loss, downtime, and income[74].

Anything put online may exist forever, even if all copies are removed, Hacking system servers may reveal critical employee data[74]. A hacker (or hacking group) may vandalize a company's website[116]. Hackers may shut a company's website, inflicting financial loss[74]. If the website is down for too long, the company may seem untrustworthy[117]. Private files, trade secrets, and intellectual property may be disclosed if a company's website or network is hacked[74]. Losing this information might slow company growth[74].

A breach costs more than replacing lost or stolen technology, investing in current protection, and strengthening physical security[74]. The company may need to alert customers and prepare for litigation. Upheaval may generate employee departures. The company may need to choose image above growth[74].

1. Security Breach Example 1

LastPass saw strange network behaviour in July 2015. Email addresses, password reminders, and authentication hashes were obtained. Hackers couldn't access encrypted password vaults[74]. LastPass protected user accounts despite a security vulnerability, any new login from a new device or IP address requires email verification or multi-factor authentication. The master password is required[74].

It is imperative for the LastPass users to protect their accounts in order to avoid being vulnerable to the hackers[74]. Master passwords should be difficult and changed often. Phishing should be avoided[74]. An attacker might phish by sending bogus LastPass emails[74]. Emails invite users to update their password through a link. The email link leads to a fake

website that steals the master password. Never click email links. Password reminders should be used carefully. Password reminders shouldn't reveal passwords[120]. When multi-factor authentication is available, users should use it[74].

If both users and service providers adopt the required tools and methods to protect users' data, the data may be safe even if there is a security breach[74].

2. Security Breach Example 2

In November 2015, Vtech's database was hacked[75]. Millions of consumers, including children, might be affected. The data breach revealed client names, emails, passwords, photos, and chat logs[75].

Hackers targeted a toy tablet[75]. Through toy iPads, consumers swapped images and chatted. The firm website did not enable secure SSL connection. The firm was suspended from the stock market despite not exposing credit card or personal identity data[75].

Vtech's hack exposed clients' personal information. Even though the corporation told consumers their passwords were hashed, hackers could decrypt them[75]. The database passwords were encrypted with MD5, but the security questions and answers were not. MD5 is vulnerable. By comparing millions of hash values, hackers may identify original passwords[75].

Cybercriminals may use the disclosed information to establish email accounts, ask for credit, and conduct crimes before the children started school. Parents' internet accounts might be compromised since many individuals repeat passwords[75]. The security breach jeopardized customers' privacy and tarnished the company's image, as shown by its suspension from the stock exchange[75].

Parents should worry more about their kids' internet privacy and demand safer goods. Manufacturers of network-connected products must secure consumer data and privacy as cyberattacks grow[75].

3. Security Breach Example 3

Equifax is a U.S. credit reporting agency. This corporation gathers millions of consumers and corporate records[76]. Customers' credit ratings and reports are based on gathered information. This might hinder loan and job applications[76].

Equifax disclosed a compromise in September 2017. The attackers exploited Apache Struts[76]. The business suspects cybercriminals obtained millions of U.S. users' personal data between May and July 2017. Personal data includes complete names, SSNs, birth dates, addresses, and other facts. There's evidence the hack impacted UK and Canada clients[76].

Equifax launched a webpage where users can check whether they were hacked and sign up for credit monitoring and ID theft protection. Criminals may create similar websites using a different domain name instead of a subdomain of equifax.com[76]. Phishing sites may utilize these sites. An Equifax employee emailed a phony link to worried consumers. This webpage was immediately removed. It highlighted Equifax's response page vulnerabilities[76].

One may wish to investigate whether the information was compromised as a concerned client. During a crisis, one may use unauthorized websites[76]. It is necessary that one should avoid giving out personal information to avoid being victimized again. Companies must protect its data from illegal access. Companies must patch and upgrade software to prevent

exploits[76]. Company workers should be taught on how to preserve data and respond to a breach[76]

Unfortunately, those whose data was exposed are the true victims. Equifax must preserve consumer data when performing credit checks since clients did not utilize its services[77]. Consumers must trust the firm to protect their data. Since both parties have the identical information, attackers may easily assume your identity[77]. In these instances, be cautious when disclosing personal information online[136]. Regularly check credit reports (once per month or once per quarter). Report any incorrect information, such as credit card transactions that one didn't make[77].

2.2.23 Types of Attackers

Attackers exploit weaknesses for personal or financial benefit. Credit cards, product designs, and everything of worth are targets[78].

1. Amateurs They're nicknamed Script Kids. Unskilled attackers who utilize the Internet. Some are interested; others wish to do harm. Simple tools, terrible results[78].

2. Hackers hack into computers or networks. Attackers might be white, grey, or black caps. White hat attackers uncover weaknesses in networks or computers[78]. Owners are notified about break-ins. Black-hat attackers exploit vulnerabilities for financial or political gain[78]. Between white and black and gray-hat attackers[178]. Gray-hat hackers may find system vulnerabilities. Grey-hat hackers may expose system vulnerabilities if it matches their goal. Some grey hat hackers publish vulnerabilities publicly[78]. White, black, and grey hat hackers are compared[78].

3. Organized Hackers These hackers include organizations of cyber criminals, hacktivists, terrorists, and state-sponsored hackers. Cyber criminals are usually groups of professional criminals focused on control, power, and wealth. The criminals are highly sophisticated and organized, and they may even provide cybercrime as a service to other criminals[79]. Hacktivists make political statements to create awareness to issues that are important to them. State-sponsored attackers gather intelligence or commit sabotage on behalf of their government. These attackers are usually highly trained and well-funded, and their attacks are focused on specific goals that are beneficial to their government[79].

2.2.24 Internal and External Threat

1 Internal Security Threats

Attacks may come from within or outside an organization as shown in the figure 2.7. An employee or contractor may accidentally or intentionally:

1. Mishandle private data
2. Threaten internal servers or network infrastructure devices
3. Facilitate outside assaults by connecting infected USB media to company computer system.
4. Malicious emails or websites may introduce malware to a network.

Internal threats may do more harm than exterior threats since internal users have direct access to the building's infrastructure. Employees also have knowledge of the company network, its resources, and its secret data[79].

2 External Security Threats

External threats might exploit network or computing device vulnerabilities or utilize social engineering.

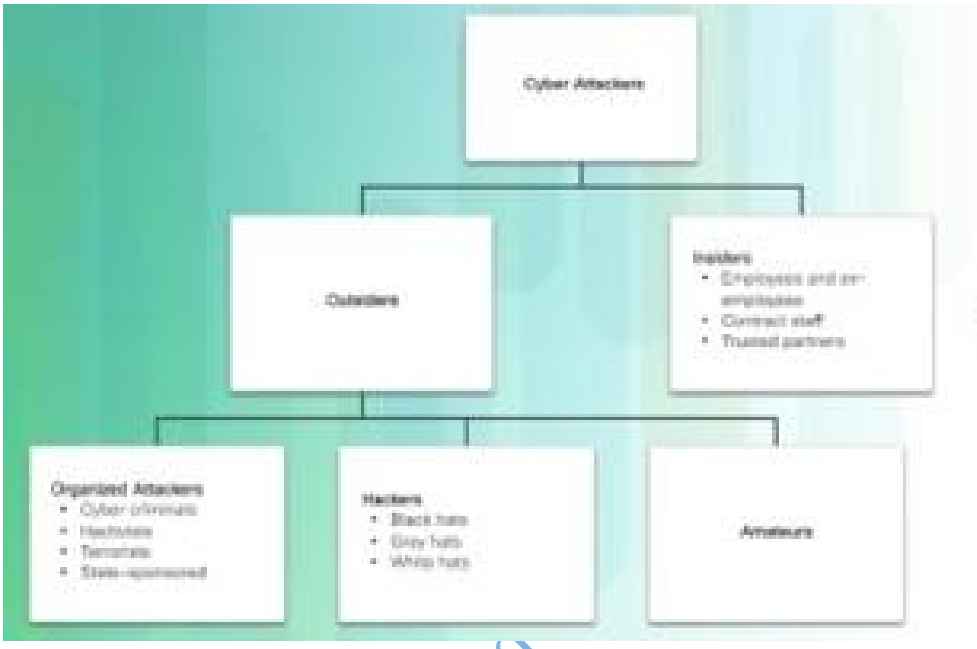


Figure 2.7 Flow Diagram of Security Threat[86]

2.2.25 The Current Threat Landscape

Always-on connections and technical breakthroughs increase the risk of exploitation. Internet of things (IoT) makes every gadget vulnerable[87]. In October 2016, Distributed Denial of Service (DDoS) attacks on DNS servers brought down GitHub, Paypal, Spotify, and Twitter[80]. This was possible due to IoT vulnerabilities. Using IoT to conduct a cyberattack is new, but its weaknesses aren't[80]. Sometime ago. In 2014, ESET detected 73,000 default-password cameras. IO Active found 7,000 vulnerable Linksys routers in April 2017[80].

What's the company's connection to home device vulnerabilities**?**

Chief Information Security Officer (CISO) should react when questioned. Because the CISO should know how personal devices influence enterprise security. Remote access, BYOD[80].

The number of remote workers is growing. Gallup reports that 43% of employed Americans work remotely using their own infrastructure[88]. Many companies allow **Bring Your Own Device (BYOD)**[80]. Most BYOD failures are due to poor planning and network architecture[80]. What connects the technologies other than the user which is the main factor that could lead to the compromise of the system. Majority of security threat that are so rampant in a cyber space today are as a result of people compromise. Old risks like phishing emails are still on the rise because they target the user's psyche[88]. Once a person performs one of them, their device is infected[80].

A spear phish campaign could start with a phishing email, which will basically be the entry point for the attacker, and from there other threats will be leveraged to exploit vulnerabilities in the system[80]. Ransomware exploits phishing emails. First-quarter 2016 ransomware payments totaled $209 million. Ransomware growth will stabilize in 2017, but techniques and targets will vary[80].

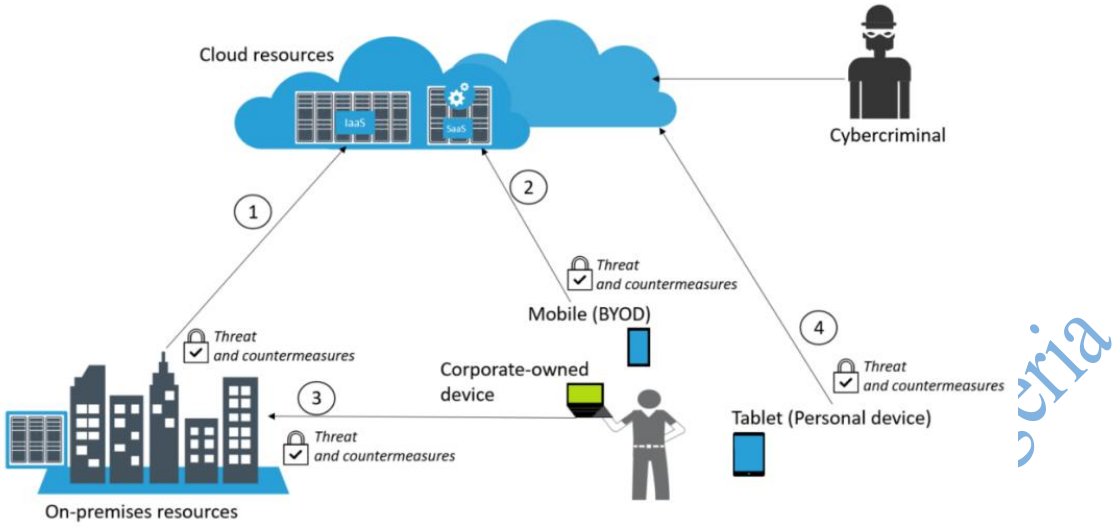


Figure 2.8 Highlights the Correlation Between These Attacks and The End User[80]

• On-premises and cloud connectivity (1)

• Cloud connectivity for BYOD devices (2)

• Interconnection of corporate-owned devices and on-premises devices (3)

• Interconnection between personal devices and the cloud (4)

End-users connect these options. Figure 2.8 shows how hackers exploit cloud resources. Cloud computing resources are a recurring issue[80]. One can't disregard cloud computing, as many companies do. As their first cloud service, most will choose infrastructure as a Service (IaaS). Some companies may use Software as a Service (SaaS). Mobile Device Management (MDM), in case (2). Military and other high-security organizations may not use the cloud. The cloud will dominate most deployments[80].

On-premise security is critical, because it is the core of the company, and that's where the majority of the users will be accessing resources. When an organization decides to extend their on-premise infrastructure with a cloud provider to use IaaS (1), the company needs to

evaluate the threats for this connection and the countermeasure for these threats through a risk assessment[80].

The last scenario (4) might be intriguing for some skeptical analysts, mainly because they might not immediately see how this scenario has any correlation with the company's resources[81]. Yes, this is a personal device with no direct connectivity with on-premise resources[81].

However, if this device is compromised, the user could potentially compromise the company's data in the following situations:

• Accessing business SaaS services from this device

• Opening a corporate email from this device

• If a user uses the same password for personal and work email, brute force or password guessing may compromise the account.

Having technical security controls in place could help mitigate some of these threat against the end user. However, the main protection is continuous use of education via security awareness training[81].

The user is going to use their credentials to interact with applications in order to either consume data or write data to servers located in the cloud or on-premise. Everything in system has a unique threat landscape that must be identified and treated[81].

2.2.26 The Credentials – Authentication and Authorization

Verizon's Data Breach Investigations Report[9], threat actors' goals and methods differ by industry. According to the research, financial incentive or organized crime prefer stolen credentials. This data reveals that threat actors are pursuing user credentials thus firms must concentrate on authentication, authorization, and access rights[81].

User identity is the new boundary, industry agreed. This requires security rules that authenticate and authorize users depending on their job and network data needs. Cybercriminals may use stolen credentials to get access to the system[82]. Having a genuine network user account will allow them to pivot and, at some time, elevate power to a domain administrator account[82]. Applying the traditional principle of defense in depth to safeguard a user's identity is still a smart method, as demonstrated in the figure below:

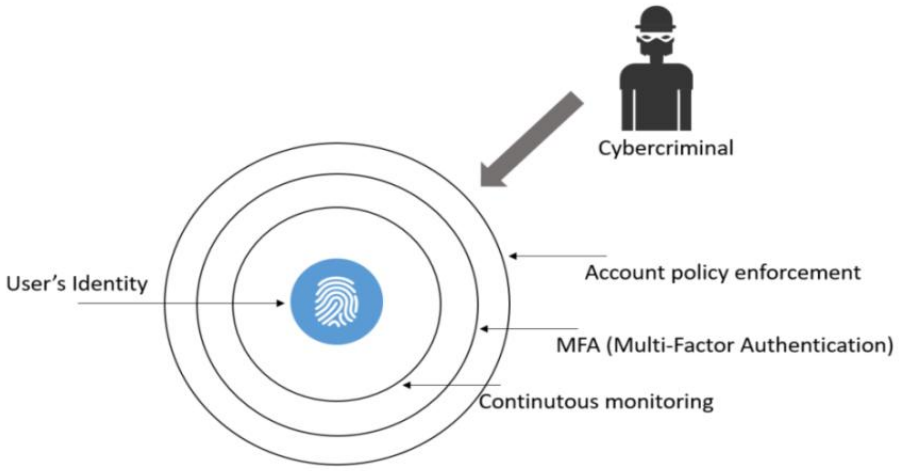


Figure 2.9 The Credentials – Authentication and Authorization[82]

Here, there are many tiers of protection, starting with account security policy enforcement, which follows industry best practices such strong password requirements[82]. MFA secures user identities. The callback feature, when the user authenticates using login and password, is popular. Authentication is successful if both factors pass[82].

2.2.27 Cybersecurity Challenges

To examine firms' cyber security concerns, data and market proof are needed. Not many industries face the same cyber security dangers[83]. This appears to be the best technique for cyber security analysts who aren't specialized in various businesses but may need to deal with them[83].

Kaspersky Global IT Risk Report 2016 (14), the costliest data breaches are caused by existing threats that are changing.

1. Malware, viruses, and Trojans
2. Inattention and unskilled employees
3. Phishing and social engineering
4. Targeted attacks
5. Crypto and ransomwares

2.2.28 Trends of Cyber Security

Cybersecurity is important in data technology. Today's biggest challenge is protecting data for safety from Cybercrimes which are rising on daily basis and ensure that no attack strikes a chord[83]. Various governments and organizations are fighting cybercrime. Cybersecurity measures still frighten many. Some cyber security trends include:

1. Web Servers

Web application attacks to steal data or spread malicious malware persist on internet-based system. Cybercriminals send code via compromised web servers. Information-stealing assaults, which often attract public attention, are also a concern. People must focus more on safeguarding web servers and online apps[83]. Cybercriminals mostly steal information from

web servers. Thus, one should always use a secure application, especially during important interactions, to avoid these defilements[83].

2. Mobile Networks

Web application attacks to steal data or spread malicious malware persist due to the vulnerability of majority of the application on mobile network. Cybercriminals send code via compromised web servers. Information-stealing assaults, which often attract public attention, are also a concern. People must focus more on safeguarding web servers and online apps[84]. Cybercriminals mostly steal information from web servers. Thus, one should always use a secure application, especially during important interactions, to avoid these defilements[84].

3. Encryption

Encoding messages so programmers can't read them. Encryption converts messages into jumbled figures[84]. It ends with an "encryption key" that encodes the message. Early encryption ensures information safety and credibility[84]. Cybersecurity issues increase with increasing encryption. Encryption is used to protect information in transit, such as on the Internet, mobile phones, and wireless radios[84].

4. APT's and Targeted Attacks

Advanced Persistent Threat (APT) is cybercrime ware and a long-term network security. IPS or web filtering can identify targeted attacks[85]. As attackers use more questionable approaches, network security must combine additional security advantages to detect attacks. To counter future threats, it is important that security protocols should be installed.

2.2.29 Role of Social Media in Cyber Security

Some people live on social media it to keep in touch with people across the globe, organize events, post photos, and remark[86]. It's replacing email and phone calls, and were swamped with the emerging technologies. As with everything online, one must recognize the risks. PCs, mobile phones, and other devices are invaluable for connecting and collaborating with the globe. Individuals may achieve this through social media[86].

Social media allows users to share ideas, images, activities, or any area of their life with others, whether they live locally or throughout the world. These networks symbolize PC security, protection, and security. As teachers use social media more, assaults rise[86]. Since most people use social networking sites regularly, fraudsters use them to steal private data.

The companies must also be quick to recognize risks, respond, and avoid a breach[87]. Individuals must take precautions, especially with social media, to prevent data loss. The ability to share information with millions of people is essential to social media's value to businesses[87]. Social media allows anybody to spread financially sensitive information and misleading information[87]. Being harmful is enough. Social media's quick dissemination of false information is an increasing risk[87]. Social media may be used for cybercrimes, yet businesses can't stop using it because of its importance. Instead, arareness should establish systems that alert them to risks so they can remedy them[87]. Organizations should comprehend this and monitor the significance of breaking down data primarily in social debates to prevent hazards. Social media contracts need specialized strategy and technology[87].

2.2.30 Cyber Terrorism

The term “terrorism” can allude to the illegal utilization of power or viciousness against people in order to threaten an administration or its residents and associations which might be to accomplish a political or a malicious site[88]. Terrorism has transformed from the conventional structure to the cyber type of innovation supported terrorism recognized as cyber terrorism. It stays vital issues of the present society[88]. Not just that the battle against terrorism is falling behind, current cybercrime assaults are ending up progressively forceful and confrontational[88.] This terrorism is the utilization of cyber word to dispatch an assault to the essential foundations that the presence of associations and countries entirely depended after that can prompt it shut down[88].

2.2.31 Components of Cyber Terrorism

Researchers in the exploratory network saw cyberterrorism as a serious threat to the security of the system. Feng in their hypothetical model recognize the five sections that T"cyber-terrorism" classified. they are; the objective of the violence, inspiration and dedication towards the mission to be accomplished when such incident occurs, impact, instruments used to dispatch such assault and attacking's, area which is natural just as the strategy for action. Knowing criminals' activity profiles offers confidence[89].

Cyber terrorism is motivated by violence/damage of some important parts of the system. Terrorists use the internet to spread diseases. An author suggested that a terrorist may cause more damage using online than with conventional terrorism[89].

2.2.32 Motivating Factor of Cyber Terrorism

Cyberterrorism is motivated by:

1. Websites' Supportive Nature

The internet is a huge medium that may help certain people join a community of interest[134]. Cyberterrorists use the website because it may refer a message to many people in a flash; they believe it is an easy way to find interested people[90].

2. Anonymity Nature of the Internet

Each evildoer seeks anonymity so they can't be identified after committing a crime. Internet is a hiding place for terrorists who may remain anonymous[90].

3. Hacking

Unauthorized entry to any "computer system" network is called "cyber murder" Many hackers employ "brute force" to try every letter, number, and picture until they acquire the password[90].

4. Computer Viruses

These viruses are distributed over a system to cause damage. These may be administrative agents, information creators, or system splitters[90].

5. Password Sniffing

"Cyber terrorists" may utilize password sniff to finish their "cyber-attack" on various nations and huge corporations to observe their collapse and control their systems. The password sniffer is software that screens, organizes, and catches the full system password[90].

2.2.33 Consequences of "Cyber Terrorism

Cyber terrorism is a unique cyber threat and assault that may harm governments and organizations. Cyberterrorism implications include:

1. Data Intrusion

Cyber terrorism may destroy information honesty so it can never again be trusted, destroying its categorization and impeding its accessibility. The growing prevalence of cyber terrorism invading organizations and countries' information has caused a tone of problems, including loss of vitals and crucial information[91].

2. The Attack on Businesses

Cyberterrorism may cost businesses billions. Terrorists may hack a bank's data and cause unauthorized access to its financial balance, causing it to lose millions of dollars and go bankrupt[92].

3. Loss of Life

Cyber terrorism has saved countless lives and damaged many homes, causing some families emotional harm. Cyber-terrorism may cause death and serious injury. It has caused assaults on PCs, networks, and aviation crashes worldwide, claiming many lives[92].

4. Consumer Trust in Doubt

The growth of every business and its support depends on the confidence that its customers have in it. Trust may see instruments that strengthen relationship and certainty between companies and clients[92].

2.2.34 Cyber Warfare

Cyberspace has become another important dimension of warfare, where nations can carry out conflicts without the clashes of traditional troops and machines[93]. This allows countries with minimal military presence to be as strong as other nations in cyberspace[93].

Cyberwarfare is an Internet-based conflict that involves the penetration of computer systems and networks of other nations. These attackers have the resources and expertise to launch massive Internet-based attacks against other nations to cause damage or disrupt services, such as shutting down a power grid[93].

An example of a state-sponsored attack involved the Stuxnet malware that was designed to damage Iran's nuclear enrichment plant. Stuxnet malware did not hijack targeted computers to steal information[94]. It was designed to damage physical equipment that was controlled by computers[94]. It used modular coding that was programmed to perform a specific task within the malware. It used stolen digital certificates so the attack appeared legitimate to the system[94].

2.2.34.1 Purpose of Cyber Warfare

Cyberwarfare aims to obtain an edge over governments or rivals.

To close industrial and military disparities, a government may penetrate other countries' infrastructure, discover defense secrets, and gather technology[95]. Cyberwarfare has the potential to devastate other nations' infrastructure and cost lives[95]. An attack might knock down a city's power grid, disrupt traffic, stop the buying and selling of goods and services, etc. There is no emergency treatment accessible. The internet could be down. Residents' everyday life may be jeopardized if the electrical system is attacked[95].

Sensitive data that has been compromised might be used to blackmail government personnel. The information may allow an attacker to pose as a legitimate user[95].

People may lose trust in the government's safety if it is unable to combat cyberattacks. Cyberwarfare may destabilize a country, impair trade, and harm residents' trust in their government without invading[95].

2.2.35 Case Study Examples

2.2.35.1 Cyber Security in E-Governance Case Study

E-Government extends government attempts to rebuild relationships with citizens. EGovernance's openness and transparency bring governments closer to their citizens[96]. Existing and future cybersecurity concerns are 21st century problems. E-Governance requires data security best practises[96]. As with security technology, rules, methods, and strategies must be established. It helps protect e-Government systems from assault, acknowledges good administrations, and establishes a backup plan. EGovernment cybersecurity requires an open private organisation[96]. These associations face coordination challenges[96].

2.2.35.2 Kaspersky Kidnapping Case

In the "highest-profile" cyber monitoring, stalking, and kidnapping case, Ivan Kaspersky was involved. Ivan Kaspersky was abducted for ransom in 2011[97]. According to Russian media, an obligated couple coordinated the scheme and enlisted their son and two pals as "muscle"[97]. Abductors followed Kaspersky and his fiancée before the seizure to determine his behaviour standards[97]. The hijackers accessed Kaspersky's Vkontakte profile, according to reports. Kaspersky begged his father for money[126]. Abductors may have exploited wireless meal delivery or geolocation[97].

2.2.25.3 Uber Case Study

Every day, in too many places, data breaches occur, but the risk isn't proportional to the number[98]. It may also rely on the company's profitability and impact on consumers or account holders. Uber had one of the biggest data breaches recently[98].

1. Impact

A recent breach compromised the personal data of 57 million Uber users and 600,000 drivers[98].

2. Details

Uber's hack reaction is a lesson for other companies. Late 2016, two hackers stole names, phone numbers, and emails. 600,000 IDs were taken[98]. Third-party cloud provider hacked Uber's GitHub. Hackers obtained Uber Amazon Web Services user data using GitHub credentials. Uber paid two hackers $100,000 to delete stolen data and keep it quiet. Uber hackers lost data[98]. US law enforcement recommends avoiding paying hackers for intrusions. Uber's tactic prompted other hackers to blackmail Netflix by threatening to disclose TV episodes. Uber agreed to pay $20 million to settle FTC charges[98]. UK, Italy, and Philippines also answered. Uber's hack is unique since it was concealed[98].

3. Uber's Plan After the Breach

Researcher was criticized for sexual harassment, underpaying drivers, and more.

4. Solutions

Cybersecurity Cyber security methods.

• Passwords and access control: Passwords and logins secure data. It's important for cybersecurity[99].

• Data Authentication: One should always check documents before sending. It should check for a trusted source and changes[99]. "Antivirus" software on devices checks these files. Virus-protection software is essential.

• Anti-virus software is a PC application that identifies, prevents, and removes malicious software, such as viruses and worms. Most "antivirus programs" include an "auto update" option that downloads infection profiles[99].

• Malware scanners check current records and archives for malicious code or malware[160]. Trojans, worms, and viruses are malware.

• Firewall: Consist of Software or hardware that monitors hackers, viruses, and worms on the Internet. All web data is routed via a contemporary firewall, which checks each communication for security flaws[99]. Firewalls must detect malware.

2.2.26 Prevention of Cyber Terrorism

The capacity to prevent cyber terrorism lies with the capacity to securely verify cyberspace[100]. Cybersecurity has an intriguing parallel to terrorism. Both are lopsided. Guaranteeing the security of information, data, and correspondence is impressively harder than hacking into a framework[100]. The attacker has an inalienable preferred standpoint in both regular terrorism and cyber-attacks. On account of state-supported attacks, the difficulties are of a lot higher greatness[100]. Governments should guarantee that their rules smear to cybercrimes and be wholly actualized and hold fast to; it is essential that the countries of the biosphere take measures to guarantee that its punitive and technical law is satisfactory to address the difficulties presented by cybercrimes[100]. The availability, confidentiality and the integrity of information in any associations are essential which endeavors must be set up to guarantee that they are exceptionally secure because it is the

significant cyber resource that makes each association stand and, in the meantime, depended upon. The information as entered by the "cyber-terrorist" is something beyond records which may incorporate messages, web applications, web pages, and just as some indispensable operating systems[100].

2.2.27 Misuse and Abuse

Cybercriminals steal information through botnets, data breaches, and manual hijacking, and hurt victims. This section discusses how cybercriminals exploit victims' internet accounts[100].

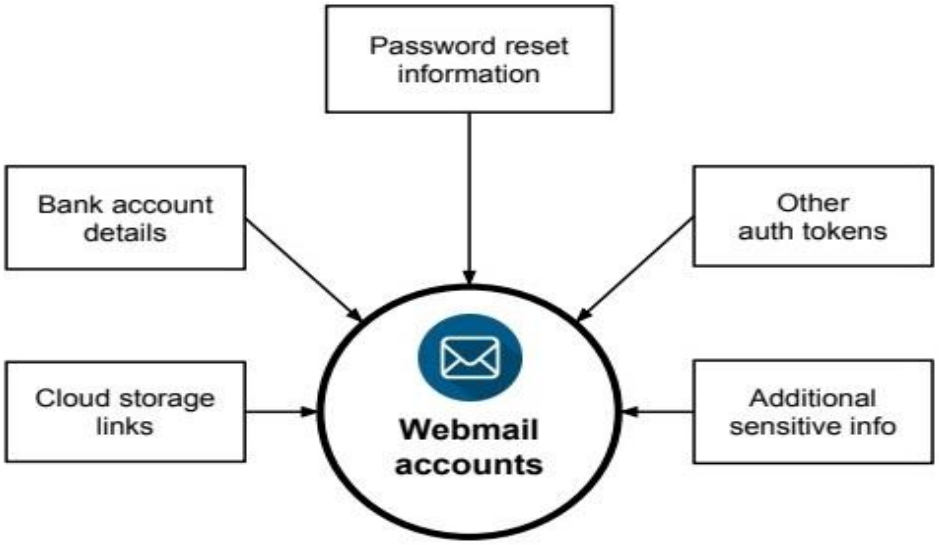


Figure 2.10 Webmail Accoun[100]

Webmail accounts collect sensitive information like other internet accounts. Cybercriminals steal and sell this sensitive data[100].

2.2.28 Information Theft and Misuse

Criminals use botnets or phishing to steal sensitive information. It was explained that virus grabs login credentials and delivers them to C&C servers. Corebot and Dridex steal

information[111]. Dridex caused $100 million in global losses in 2015[111]. This underscores the damage caused by information theft and the need for more research to minimize it. Cybercriminals hoard stolen sensitive information for later use or sell it on dark marketplaces, underground forums, and paste sites[111]. Illicit applications of stolen information include spamming (using stolen authentication credentials), spear phishing, and blackmail[111].

2.2.29 Spam

Spam, or unwanted message, has long plagued internet services, including webmail and social networks[112]. These services attract numerous users and acquire personal data. This draws hackers seeking sensitive data[112]. To access this goldmine, they use botnets to distribute spam with malware payloads to unsuspecting victims[112]. Sometimes they send spear phishing communications[112]. Some spam mailings include innocuous newsletters, commercial offers, and other non-malicious material. Researcher said spam protection efforts by the security community are disproportionately expensive[112].

2.2.30 Scams

Cybercriminals now seek financial gain rather than harm[66]. Cybercriminals defraud unsuspecting individuals to make illegal money. This recalls 419 frauds. 419 scams are specified under Section 419 of the Nigerian Criminal Code[113]. A traditional 419 scam begins with a communication from a fraudster to a prospective victim, generally outlining imaginary riches. After gaining the victim's confidence, the fraudster asks for a "small" amount to complete a "prize." The fraud is over after the victim pays. Scammer stops contacting victim. Even if the scammer restarts contact, it's generally to get more money[66]. These manual procedures depend on the victim's avarice or sympathy to succeed.

Other prevalent internet frauds include dating scams that use dating websites. Cybercriminals put up false accounts to entice lonely people. Scammers acquire victims' confidence and demand money for visa processing and travel tickets. Scammers want pricey flower baskets[66].

Before the Internet, scam letters were sent. 419 scam mails are being transmitted en masse by email and fax. Online scams cause victims financial and psychological harm and aren't well understood[66].

2.2.31 Cyberbullying

Social networks have expanded from online venues where users communicate with friends, family, and strangers to huge platforms and ecosystems[67]. These platforms are heavily used by many people across the world due to its level of importance in reaching out to friend and relatives and making new friends irrespective of the location where the person resides. Instagram superstars with millions of followers earn a career from social media. Social networks attract toxic behaviour like cyberbullying. Previous research examined cyberbullies, victims, and onlookers. Other research examined cyberbullying in social networks, toxic online forums and gaming groups[67]. Anonymous online behaviour has been studied. Harmful online networks don't contain toxic behaviour. There's evidence that 4chan.org coordinates assaults on other services[67].

2.2.32 Detecting Malicious Activity

Researchers are interested in fraudulent online account activities. This section reviews research on detecting and mitigating harmful internet behaviour.

2.2.32.1 Manual Hijacking

Some researchers studied manual account hijacking, not botnet hijacking. Manual hijacking is rare, and phishing is how manual hijackers get user passwords. The research shows how fake account credentials help comprehend harmful behaviour[68]. Other research has used phony credentials and documents to examine harmful activities. Cybercriminals prefer compromised accounts over phony accounts because harmful behaviour is harder to identify[68]. COMPA identifies fraudulent conduct in online social networks by constructing statistical models of user behaviour. Deviations from this pattern help identify hacked accounts. New methods were created by cyber security researcher to identify spear phishing attempts based on behavioural modelling of senders[68]. The technology searches for aberrant email sending and writing behaviours rather than reviewing email content for suspicious terms. Some used sandboxed phishing kits to study the phishing environment and life cycle[68].

2.2.32.2 Understanding Spam

Some researchers analyzed spam Twitter accounts. They discussed spammers' methods. Most spam accounts use unsolicited mentions and hashtags to attract audiences outside their social relationships. They also found a social spamming environment (including affiliate programs)[69]. The research did not offer detection measures to find spammer activities, although noting that Twitter's present protection systems are mainly ineffective. It examined social spam employing 900 honeypot accounts and offered a Facebook and Twitter spam tool. Also, researched social spam and used machine learning to identify real YouTube users from video spammers and content marketers. Also, Spam was by promoting honeypot email addresses online[69]. The research revealed the connections among email

harvesters, spammers, and botmasters[116]. Other also analyzed 16 C&C servers of Pushdo/Cutwail botnet. Other research examined network-level spam detection, statistical/machine-learning techniques, and the underground spam ecology[69].

2.2.32.3 Detecting Fake Accounts

Spam is fueled by fake accounts. A researcher suggested that using click event patterns to recognize fraudulent accounts, known as Sybils, in online services[69]. Using clickstream models, they taught machine learning to recognize bogus accounts[69].

2.2.32.4 Defeating Information Theft

Research in some of the previous literature how Torpig botnet was hijacked for ten days by taking advantage of weaknesses in communication protocols of the botnet[69]. Their technique of sinkholing all bot data delivered to the C&C server they hacked are identical and his colleagues explored P2P content privacy by distributing honey files with honey account credentials. They observed downloads and determined that attackers who downloaded honey files wanted to profit from private data. In this thesis, Fake account credentials were posted in crucial areas for fraudsters to utilize. This thesis focuses on online accounts, not P2P networks[69].

2.3 Review of the Empirical Studies

2.3.1 Cyber Security

An empirical review in cybersecurity involves conducting research and analyzing empirical data to better understand and address various aspects of cybersecurity. It focuses on examining real-world data, observations, and experiments to gain insights into the effectiveness, impact, and implications of cybersecurity practices, technologies, and

strategies. Here are some examples of areas where empirical reviews in cybersecurity can be conducted:

1. Effectiveness of security controls: An empirical review would evaluate the effectiveness of various security controls, such as firewalls, intrusion detection systems, or encryption, by analyzing real-world incidents, data breach reports, or conducting controlled experiments. This helps determine their actual ability to mitigate threats and protect systems and data.

2. User behavior and cybersecurity: Understanding user behavior is crucial in addressing human-centric cybersecurity threats, such as social engineering attacks. An empirical review would involve studying user attitudes, awareness, and decision-making processes related to cybersecurity through surveys, interviews, or behavioral experiments. This research can inform the development of effective awareness programs and user-focused security measures.

3. Vulnerability analysis: Empirical reviews in vulnerability analysis involve analyzing historical data and conducting scans or penetration tests to identify common vulnerabilities in software, networks, or systems. This research helps prioritize patches, improve secure coding practices, and develop strategies for vulnerability management.

4 Incident response effectiveness: Evaluating the effectiveness of incident response measures through empirical research involves analyzing real-world incidents, response times, impact assessments, and lessons learned. This research helps refine incident response plans, identify areas for improvement, and assess the effectiveness of incident response teams and technologies.

5. Risk assessment and management: Empirical reviews in risk assessment and management involve analyzing historical data, threat intelligence, and vulnerability assessments to quantify and prioritize risks. This research helps organizations allocate resources effectively, implement appropriate mitigation strategies, and make informed decisions regarding cybersecurity investments.

6. Cybersecurity education and training: Assessing the effectiveness of cybersecurity education and training programs requires empirical research. This involves evaluating knowledge retention, behavioral changes, and skill development among cybersecurity professionals or end-users through surveys, assessments, or performance evaluations. The findings can guide the development of more effective training programs.

7. Cybersecurity economics: Empirical reviews can be conducted to analyze the economic aspects of cybersecurity, such as the costs of breaches, the return on investment for security measures, or the impact of regulations on cybersecurity practices. This research helps organizations make informed decisions regarding cybersecurity investments and policy-making. By conducting empirical reviews in cybersecurity, researchers and practitioners gain valuable insights into the real-world effectiveness, impacts, and implications of various cybersecurity practices and strategies. This evidence-based approach enables informed decision-making, the development of more effective security measures, and the advancement of the overall understanding and maturity of the field.

2.3.2 Deception Technology

An empirical review of deception technology in cybersecurity involves assessing real-world data and evidence to evaluate the effectiveness of using deceptive techniques as a defensive strategy. Here are some key empirical findings related to deception technology: **1.**

Detection and early warning: Empirical studies have shown that deception technology is effective in detecting cyber threats, including advanced persistent threats (APTs) and insider attacks. By creating decoy assets that appear authentic and valuable to attackers, deception technology can attract and alert security teams to potential malicious activities, providing early warning and reducing the dwell time of attackers within the environment. **2. Enhanced threat intelligence:** Deception technology can yield valuable insights into attacker tactics, techniques, and procedures (TTPs). Through the analysis of attacker interactions with decoy assets, organizations can gather detailed information about specific attack vectors, tools, and motivations. This knowledge can then be used to improve threat intelligence and develop more effective mitigation strategies.

3. Reduction in false positives: Deception technology has been found to significantly reduce false positive rates compared to traditional detection methods. By focusing on the deceptive assets, security teams can differentiate between genuine attacks and false alarms, saving time and resources typically spent investigating false positives.

4. Active defense capability: A key advantage of deception technology is its ability to engage with attackers actively. By presenting deceptive elements, organizations can mislead adversaries and divert their attention away from critical assets. Active engagement can help delay or prevent successful attacks, buying time for incident response actions or enabling the identification and tracking of attackers.

5. Insider threat identification: Deception technology can also aid in identifying insider threats. By using decoy assets specifically designed to appeal to insiders, organizations can monitor for unauthorized or unusual interactions, detecting potential malicious behavior or policy violations from within the organization's network.

6. Compliance validation: Deception technology can assist in meeting compliance requirements and validating the effectiveness of security controls. By deploying decoy assets that support specific compliance standards, organizations can test the response of their security measures and identify any potential vulnerabilities or gaps in compliance.

7. Deployment considerations: Empirical studies highlight the importance of careful planning and deployment of deception technology. Proper configuration of deceptive assets, regular updating of decoy content, and alignment with an organization's network infrastructure and security operations are crucial for maximizing the effectiveness of deception technology. In conclusion, empirical evidence supports the effectiveness of deception technology as a defensive strategy in cybersecurity. It offers improved threat detection, enhanced threat intelligence, reduced false positives, active defense capabilities, insider threat identification, and compliance validation. However, organizations should consider specific deployment considerations to ensure optimal utilization and avoid potential drawbacks associated with using deception technology.

2.3.3 Honeypot

An empirical review of honeypot systems involves evaluating real-world data and evidence to assess their effectiveness in detecting and understanding cyber threats. Here is an overview of key empirical findings related to honeypot systems:

1. Detection capabilities: Empirical studies have shown that honeypot systems are effective in detecting various types of attacks, including network scanning, malware infections, and unauthorized access attempts. By presenting attackers with seemingly vulnerable targets, honeypots attract their attention and provide an opportunity to collect information on attack techniques and methodologies.

2. Early warning system: Honeypots can serve as early warning systems, providing organizations with timely alerts when attackers interact with them. This early detection allows security teams to respond quickly and prevent potential damage to critical systems or data.

3. Attack analysis and understanding: Honeypots are valuable tools for analyzing attack patterns and behavior. By capturing and analyzing the activities performed by attackers within the honeypot environment, organizations can gain insights into their tools, tactics, and motives. This information can be used to improve threat intelligence, develop effective countermeasures, and enhance overall cybersecurity strategies.

4. Insider threat detection: Honeypots can also help identify insider threats within an organization. By monitoring the activities of authorized users within the honeypot environment, organizations can detect any potentially malicious behavior or policy violations.

5. Deception effectiveness: Empirical studies have demonstrated that deception techniques used in honeypots can effectively mislead and confuse attackers. By presenting attackers with realistic-looking targets and enticing them to interact with deceptive elements, honeypots can divert their attention from actual production systems, reducing the likelihood of successful attacks.

6. Compliance validation: Honeypots can be used to assess the effectiveness of security controls and compliance with regulatory requirements. By monitoring for unauthorized access attempts or interactions with sensitive data within the honeypot environment, organizations can identify potential vulnerabilities or compliance gaps that need to be addressed.

7. Limitations and considerations: While honeypot systems offer numerous benefits, there are some limitations and considerations to be aware of. Honeypots need to be carefully deployed and monitored to avoid their misuse or becoming an attractive target themselves. Additionally, the use of honeypots should be aligned with legal and ethical guidelines to ensure the responsible and legitimate use of deception techniques. In conclusion, empirical studies have consistently shown the effectiveness of honeypot systems in detecting and understanding cyber threats. They serve as valuable tools for early threat detection, attack analysis, and gaining insights into attacker behavior and techniques. However, organizations should carefully consider the implementation, monitoring, and ethical considerations associated with using honeypots to maximize their benefits and ensure a responsible cybersecurity approach.

2.3.4 Exploiting the Outcome of Honeypot Work

Security experts have employed honeypots since. Honeypot data analysis requires understanding of network protocols, applications, hardware, operating systems, and user management. This requires network security management[73]. Despite their promise, honeypots' limitations have impeded network security. Much has been done to automate and simplify raw data processing and integrate intrusion detection, data mining, expert systems, AI, and game theory[73]. This section summarizes honeypot innovations.

Cooke examined honeypot bot risks. Cooke's analysis on bots found that recent advancements in bots pose major dangers to honeypots. Cook constructed and set a honeypot to detect assaults from bots, often two at once. These data show that contemporary bots are quite active and powerful, and that honeypots are needed to defend and control them. Cooke recommends super honeypots. Cooke created the mega honeypot

as follows; First, misleading honeypots that bots may infect are created. Super-honeypots monitor misleading honeypots to catch, learn, and prevent bots[73].

Mohammed created a Double Honeynet with Principal Component Analysis (PCA) to enhance signature creation for polymorphic worms[73]. The predicted system hooks a worm in the first honeynet, allowing it to infect the next[73]. The worm may travel between the two honeynets and evolve. Every kind of worm is caught as it replicates across honeynets. Principal component analysis is used to produce a signature that may be used to find polymorphic worms using IDS[73].

Author researched the usefulness of utilizing honeypot data to discriminate worms, bots, and irregularities created by human network administrators. Temporal source counts, arrival window modification, inter-arrival distribution analysis, destination-net scan foot printing, first destination preference, source-net dispersion analysis, per-source scanning profiling, and source lifespan analysis[73]. Researcher utilized this strategy to deduce the shape of each questionable honeypot instance. Researcher verified their conclusions to confirm their methodologies[73].

2.3.5 Novel Kinds of Honeypots

To adapt to the latest changes in Internet-based services, such as advertisements related to current network applications, general acceptability of wireless connection devices, introduction of high-speed subscriber link technologies to every internet user, and differences in users' demography in terms of cultural norm, tradition, and legal systems, several new types of honeypots have been suggested and introduced[51]. This section explains current honeypot innovations[74].

Adachi introduced BitSaucer. It's a hybrid low- and high-interaction honeypot system designed to reduce resource needs in low-interaction honeypots and replicate complete replies in high-interaction honeypots[74]. The proxy running on every host generates virtual hosts and redirects network traffic. Every virtual host imitates a whole system in high-interaction honeypots, while on-demand solicitations of such virtual hosts limit resource consumptions. High-interaction honeypots are automatically activated only when network traffic requiring them arrives at a host[74].

Webb constructed a social spam honeypot. Social spamming in cyber security involves sending social networking users spam messages that may lead them to hazardous websites[125]. Honeypots examine MySpace spam invitations. Webb built fictitious MySpace profiles for each fake page[74]. The honeypot waited for spammer friend invitation requests, collected their profiles, and noted their origin network addresses protocol for later research. Midwestern states were the most popular spamming targets, whereas California had the most spam profiles. 60% of spam profiles had identical "About Me" text. Thousands of URLs and redirections are used to send spam profiles to a few websites. Webb found different temporal trends in spam profiles[74].

Rowe constructed bogus honeypots to stop intruders from production hosts. Fake honeypots look and act like real ones. False honeypots dissuade attackers by revealing themselves. Attackers avoid honeypots[74]. Very obvious phony honeypots may assist attackers recognize them as fake, whereas subtle ones won't[74]. This has led to the construction of a mathematical model to maximize the effect of fake honeypots, using

parameters such as the possibility of a system being a honeypot, an attacker's profit from compromising a production host, and the cost of compromising a host[74].

2.4 Theoretical Review of Honeypots System

Honeypots are deception tools that detect, deflect, and counter. It may be used alone or alongside an IDS/IPS. Level of engagement, deployment environment, resource type, services, flexibility, and implementation may define honeypot characteristics[74]. More study and development in this subject have led to the invention of CAPTCHA, Intrusion Detection Systems, and Honeypots. Research and production honeypots provide system security by enticing hackers in various ways. Auto-responsive honeypots help avoid denial-of-service assaults[74]. Since honeypots were introduced in 1990, much has been accomplished till 2020. Honeypots are trending in commercial and research departments[74]. Initially, there were many honeypots for kali Linux and few for windows, but Java made it possible to develop user-friendly honeypots in windows[74]. With additional investigation, honeypots took on new forms, such as using game theory to influence hackers[126]. Honeypots may be deployed with a firewall in front, behind, or both. Honeypots may also be integrated with network log management tools like Wireshark[74].

2.4.1 History of Honeypot

In 1991, "The Cuckoos Egg" and "An Evening with Breford" were published, launching the honeypot concept. Clifford Stoll wrote "The Cuckoo's Egg" about catching a computer hacker looking for secrets in his company's systems. Bill and his buddies described their experiences capturing a hacker in "An Evening with Breford"[75]. These two newspapers started Honeypots. Deceptive Toolkit, the first honeypot, debuted in 1997. Cybercop Sting was the first commercial honeypot in 1998. In 2001, honeypots were employed worldwide

to thwart attacks. Since then, honeypot research has increased. Today's honeypots utilize various software or models to capture hackers and their network activity[75].

Honeypot is the first deception technique accordingly, IT security researchers began utilizing them in the 1990s to trick hostile actors into interacting with a fake system. Honeypots collect and analyze malevolent actors' activity. They didn't identify threats. Since honeypots were established, deception technology has improved[75].

Honeypots aren't useful at detecting malicious actors since they're restricted in reach and simple to identify[75]. Hackers immediately realize they're fake. Today's deception technology promises early and efficient threat detection. Deception must go beyond the honeypot to attain its full power[75].

Anagnostakis et al. note honeypots' logistical challenges. They're hard to build and manage, so security teams can only deploy a limited number. They're never enough to identify dangers. Any usefulness a honeypot technique has for detection is reliant on the expectation an attacker would stumble across it[75].

As technology evolves, so do cybercriminals emerge. They got the honeypot trick long ago. Experience, crowdsourcing, and freely accessible technologies assist attackers differentiate honeypots from actual systems[75]. Deceptions must be inevitable, undetected, and inescapable for successful detection. These aren't today's honeypots. Originally, IT researchers used honeypots. They were meant to let the defense see assaults[127]. Threat research still uses them. They help with forensic analysis, threat hunting, and malicious behaviour responses[75].

2.4.2 The Concept of Honeypot

Honeypots are computer systems meant to "trap" hackers. It's a network decoy that mimics a genuine system. It distracts attackers who attempt to get illegal network access. Honeypots detect and learn from threats to enhance security[75]. Honeypot logs harmful activity and learns new threats. Honeypots may be used to watch a user's actions. Figure 2.11 depicts honeypot construction. Research and production honeypots exist[75].

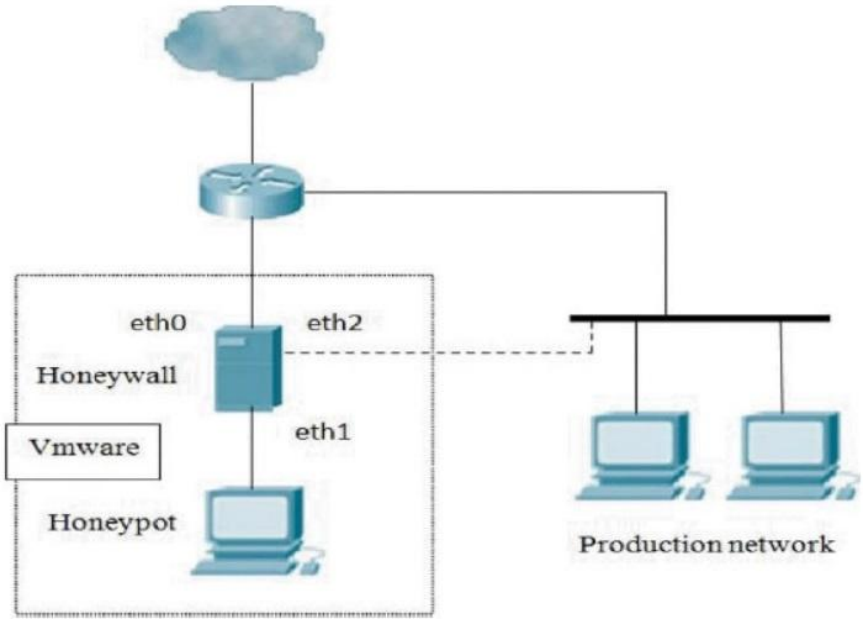


Figure 2.11 Basic Architecture of Honeypot[75]

2.4.3 Honeypot System

Honeypots are network decoys that lure attackers. Honeypots mimic real computers by executing services and applications with open ports[76]. Honeypots mimic the real system to confound and monitor attackers without endangering production servers or data. Honeypot technology enhances security. Honeypot detects network attacks. A honeypot system can recognize and redirect attack activities[76]. This system captures intrusion data to document attacker activities. It also assesses the attack's degree, tools, goal, and infiltration tactics to

gather evidence and take legal action. Honeypot systems are meant to entice hackers and track intruders[76]. Honeypot may be a computer simulation of a known vulnerability or a service computer, can mimic a range of operating systems and their characteristics, or simply a standard operating system, and only by specific processing can be a comprehensive record of the attacker's assault[76].

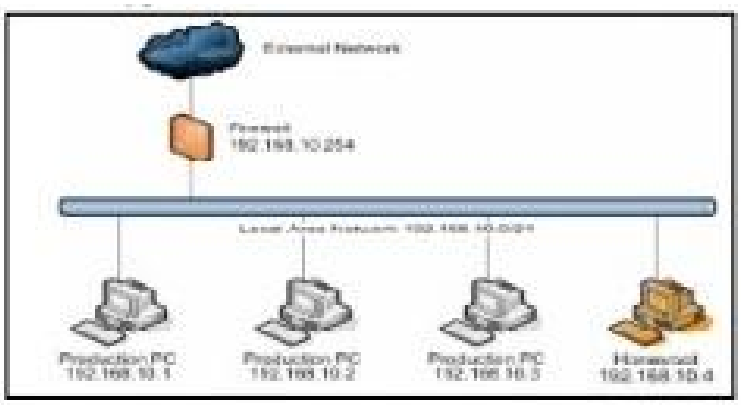


Figure 2.12: Deployment Scenario of a Single Honeypot[76]

It's a resource for unauthorized access. Honeypots are valuable because attackers abuse them[76]. Honeypots are physical and virtual. A physical honeypot is a machine with its own IP address, whereas virtual honeypots are simulated on actual computers - the VM's TCP/IP stack is meant to seem like a real system. Files and internet accounts may potentially be honeypots. Honey files are bait files that activate alarms when opened, yet they seem normal. Honey files are placed in user file locations together with bait, such as passwords.xls or account details.txt (to lure attackers). Attacks on honey files are documented[76].

In the 1980s, Stoll tracked down a German hacker who hacked LBNL's computer network. Stoll used a honeypot to do this. In the 1990s, Cheswick tricked a hacker into thinking they obtained password files and sensitive assets on an AT&T gateway computer. Advanced

honeypots are new. Honeypots have been used to research wild malware, penetrate botnets, and monitor social spam[76]. Spritzer said honeypots help reduce insider risks. This validates the use of honeypots to research harmful activities targeting online accounts and people[76]. It's a network fake to lure attackers in to the system[129]. Honeypots are meant to resemble genuine computers, running complete services and applications with open ports. Honeypot mimics the genuine system to confuse attackers and monitor intruders without risking production servers or data. Honeypot technology supplements existing security and protection methods. Honeypot detects and reacts to network intrusions[76]. A honeypot system may identify attack activity and divert it to a controlled environment. This system captures intrusion data to document attacker activities[76].

Network honeypots are resources. It operates as a host to attract attackers, but its main goal is to be attacked and studied across the network. Its fake data seems legitimate, giving the attacker the impression it's a real host[76]. Software operating in the background stores the network connection between the attacker and honeypot host to record the attacker's behaviour. Analytical approaches were employed to analyze the data or attacker's activity[76]. Induced, Deceived, and Analysis are the three honeypot modules[76]. All three parts are functional. Module induces attackers to honeypots. Fooled mimics a honeypot's database to trick an attacker. Analysis affects both modules' activities. Honeypot flowchart[76].

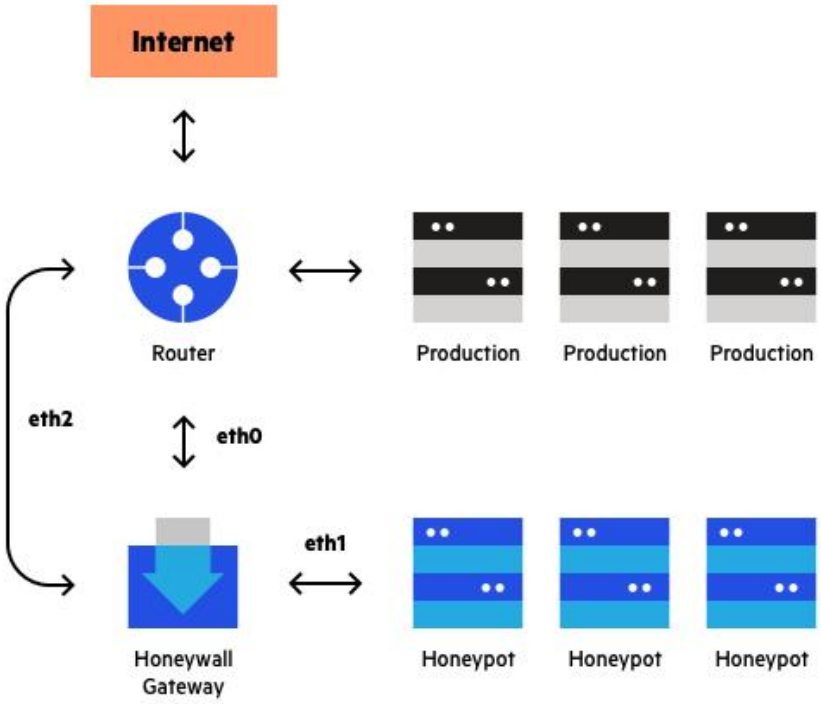


Figure 2.13 Conceptual Overview of Honeypots[76]

2.4.4 Types of Honeypots

There are two major ways through which a successful Honeypot can be designed and deployed. These are Production and Research honeypot.

2.4.4.1 Production Honeypot

A researcher said a production honeypot protects an organization and mitigates risk. Honeypots mimic the company's production network. Attackers use them to disclose production network weaknesses. Identifying vulnerabilities and informing administrators of assaults may minimize intrusion risk[77]. It's put in the production network alongside other servers like the firewall to boost security. Honeypot production reduces infiltration threats and boosts organization security[77]. Research honeypots are more complex than production ones[130]. Construction and deployment are simpler. Researcher detected assault patterns

however they gave little information on the attackers. One may understand where system attackers are coming from and what vulnerabilities they're employing, but not who they are, how they're organized, or what tools they're using[77].

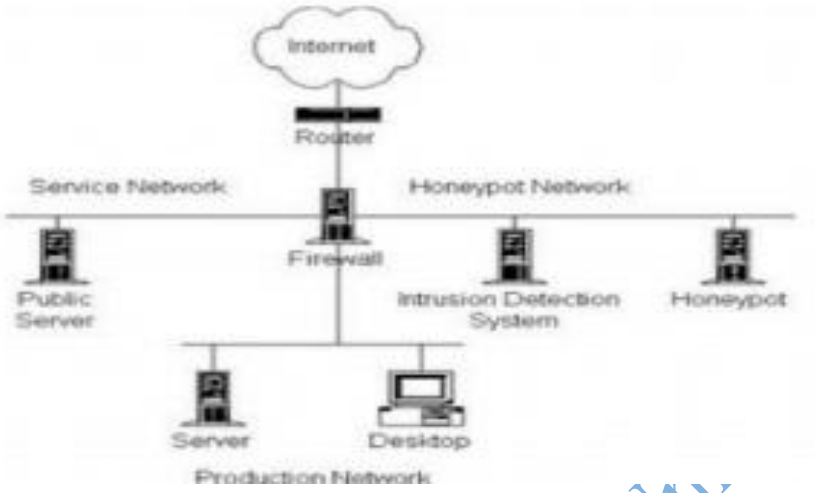


Figure 2.14: Production Honeypot[77]

2.4.4.2 Research Honeypot

Zhang's Real operating systems and services are used in research honeypots. It's often used to learn about blackhats. They're risky and gather information on future attack strategies[130]. It shows assault kinds more accurately and helps organizations investigate and guard against dangers. Honeypot research is harder to implement and manage[9]. Research, military, and government agencies utilize them[77]. Honeypots help researchers examine cyber dangers. Attackers may be seen and recorded as they breach Researcher Honeypots' intelligence gathering which is one of its best features of honeypot system. Researcher divided honeypots into three classes: low-interaction, medium-interaction, and high-interaction[77].

1. Low-Interaction Honeypot

Familiarity Honeypots don't let attackers access genuine services, apps, or OS. It produces mimicked services, operating systems, and Network protocols that an attacker may use. It's simpler to install and manage, and the danger of honeypot compromise is lower. This sort of Honeypot gathers less information since intruder contact is minimal. Low-interaction examples Spectar's Honeypot[77]. Low-interaction honeypots don't provide remote OS login[77]. They simulate a function or service in the current system, limiting attackers to a constrained region. A low-interaction honeypot monitors a port for analogue services[77]. Low interaction honeypots imitate network services such as FTP, SQL, Web, SSH, and others on predefined ports.

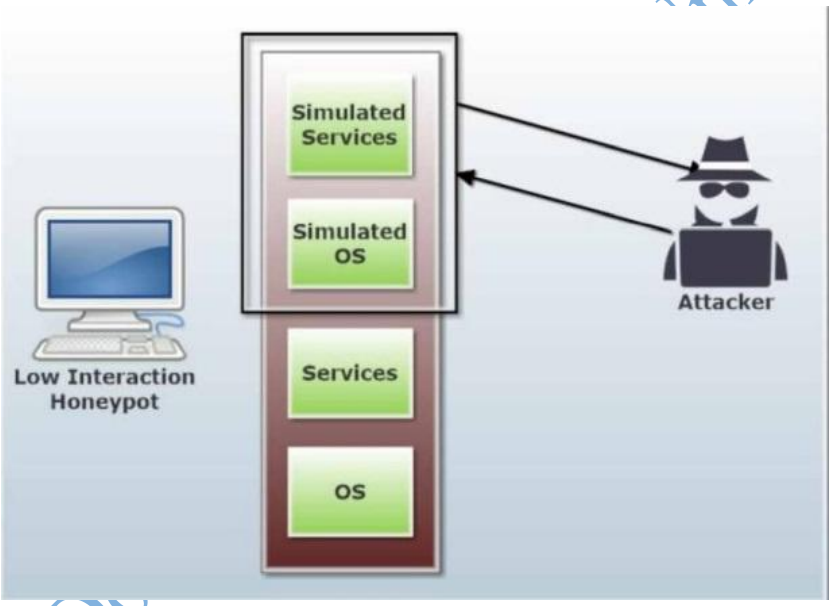


Fig 2.15 Low Level Interaction[77]

2. Medium–Interaction Honeypots

It's a hybrid honeypot that operate between the low interaction and high interaction honeypot. It performs within the medium. It lacks an operating system like Honeypot, but

phony services are more complex, enhancing data collection. By not delivering OS and services, it collects enough data[78]. Examples: mwcollect, Honeytrap, Nepenthes.

Medium-interaction honeypots give the attacker with a better OS illusion[78]. Low-interactive honeypots may be more concealed and capture more information. They engage with intruders more effectively than low-interaction honeypots[78]. This honeypot system emulates a service, making attackers believe they're assaulting the genuine OS. It allows for high data collection but raises intrusion risk mwcollect, nepenthes, honeytrap[78].

3. High-Interaction Honeypots

Engaging Honeypot allows intruders to access genuine services, apps, and OS in order to allow the attackers intelligent information to be collected. These are more complicated to install and manage, which enhances the Honeypot's vulnerability[79]. Example: Sacrificial Lamb, Spam Honeypots, Instrumented Systems.

Spitzner highlighted that high interactive honeypots use a genuine operating system to fool attackers. Real operating systems and apps must be deployed[79]. High-interactive honeypots let attackers perform genuine OS commands. All activities may be tracked and evaluated, thus there are many opportunities to gather data. Any system malfunction may allow a hacker to influence the whole OS, attack other systems, or intercept application conversations[79]. High-interactive honeypots find flaws or exploits. Honeypots facilitate zero-day attacks. Examples: Sebek.

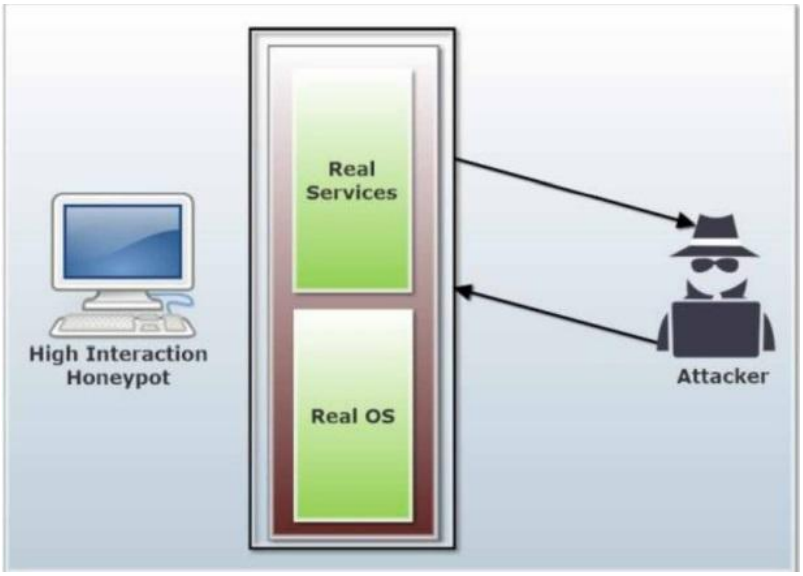


Fig 2.16 High Level Interaction[79]

The different between High level and low-level interaction can be found in Appendix VII while the difference between medium and low-level interaction honeypot can be found in Appendix VIII.

2.4.5 Deployment of Honeypot System

Honeypot can be deployed in anyplace on the server. Depending on the service, it may be utilized online or offline.

1. Internal Placement

Honeypots and firewalls are placed within the Network. Prior warning system deployment is ideal for detecting foreign exploits and internal risks[80]. Internal Honeypots make data control inside the same network difficult. These compromised systems may attack other lawful systems. This can be remedied by building a firewall and utilizing low-interaction honeypots[80].

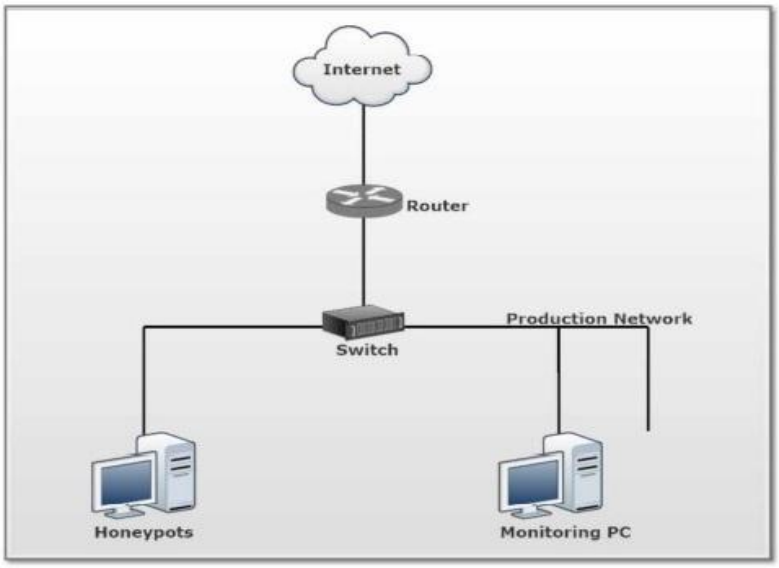


Figure 2.17a Internal Placement of Honeypot System[80]

2. External Placement

In this case, honeypot isn't protected, it allows honeypot and production PCs to share IP. Honeypot gets a public IP address when more than one is needed. Most users merely plug in and watch the honeypot. Defend honeypot if attacker hits other targets, but specifics matter[81].

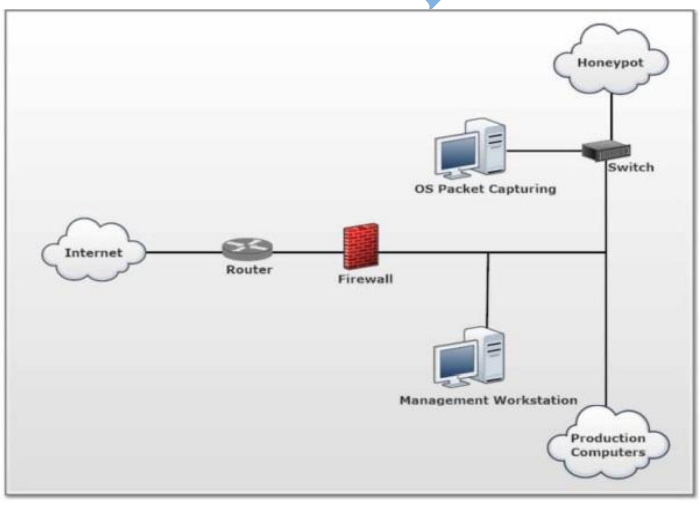


Figure 2.17b External Placement[81]

2.4.6 DMZ Placement of Honeypot System

Demilitarized Zone (DMZ) The positioning is ideal. Figure 2.18 shows the Honeypot's positioning. It forwards a duplicate of each packet from one network switch port to another[131]. Many organizations need DMZ positioning to deploy their honeypot system to function very well[81].

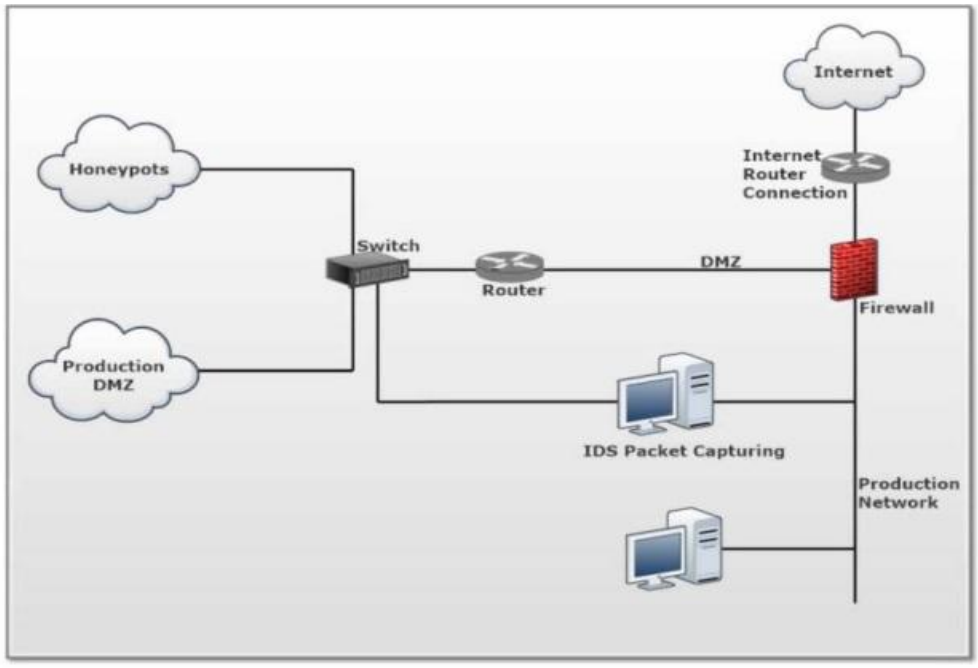


Figure 2.18 DMZ Placement of Honeypot System[81]

1. Honeynets

Honeynets is a connection of several honeypot that collect external and internal danger information[82]. Figure 2.19 shows a honeynet composed of several honeypots. The goal is to establish a tightly regulated network architecture that can capture malware, viruses, and attackers. It's a Honeypot extreme[82]. It's a network of high-interaction Honeypots, so an attacker may interact with genuine OS and services. By implementing this, a large quantity of information is acquired[82]. Honeywall stops outgoing traffic so a hacked machine can't

launch attacks. Honeywall analyses incoming and outgoing traffic. In "Know your adversary," honeynet was used to acquire information. HoneyNet Project (2003a) gathers information on credit card theft, HoneyNet Project (2003b) tracks Italian hackers tunnelling IPV6 over IPV4[82]. Incoming and outgoing packets are monitored[82]. The packet is then analyzed. Honeynet data may be used to mitigate. A researcher noted that honeynet and honeypot privacy issues is something that require attention from the cyber security specialist [82]. Honeynet's monitoring captures data. IP address is personal data, hence it should be collected and sent separately[82].

A honeynet has one or more honeypots, which are data records, files, or unused IP addresses[82]. Physical honeypots or virtual ones using VMware, XEN[82]. Honeywalls are crucial to honeynet architecture. Gateways isolate honeypots. Honeywall routes all honeypot traffic. This layer 2 gateway should be untraceable to honeypot users. Figure 2.19 illustrates it. Three-port Honeywall. These bridges (eth0 and eth1) isolate the honeypots. Third interface (eth2) includes remote IP stack[82].

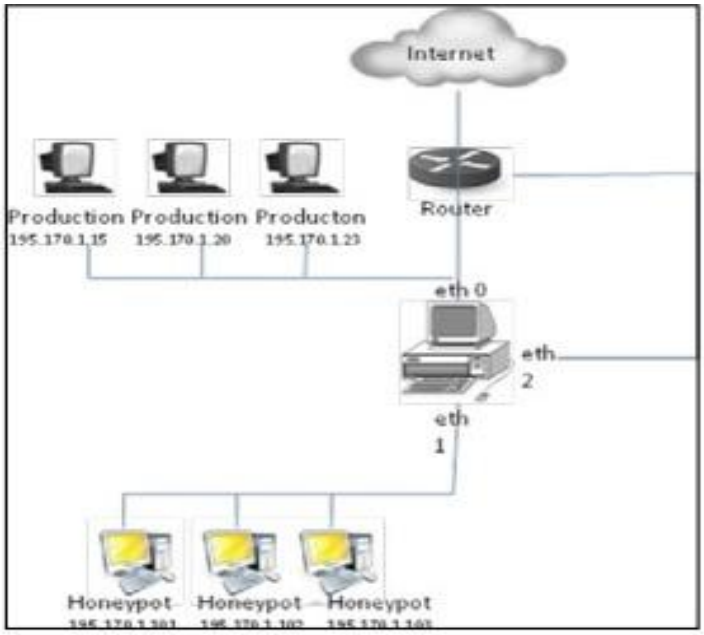


Figure 2.19: Deployment of Honeynet[82]

2. Honeytokens

It's an ingress honeypot implementation. Excel, spreadsheets are digital entities. Honeytokens are similar to Honeypots but should not be interacted with. Unofficial or hostile communication is frowned upon[83]. This approach may be utilized to catch insider threats in an organization by installing it where user identification can be checked[83]. Honeytoken communication or value modification is an easy-to-catch sign of an attacker. These are very adaptable since they can adapt to any circumstance or setting. It's in the lead since it doesn't need physical systems like conventional Honeypot and because it's inexpensive[83]. The authors employ network traffic and raw logs to analyze activity and determine insider risks in a company. This semi-comprehensive system identifies aberrant attacker activity and delivers alerts[83]. This strategy works with high-rate network sessions[131]. This technique ignores network transfers from several hosts to a single host. Bercovitch.et.al suggested a three-phase automated honeytoken creation mechanism. Rule

extraction gets distinct rules from the database, honeytoken production generates honeytokens based on the retrieved rules, and similarity rating is derived in the final step[83]. Shabtai did a two-part investigation. In the first section, the author employed a general way to produce honeytokens that resemble database tokens and real-world data items. Next, the author performed research to show that user behaviour does not alter when honeytokens are implanted in the database[83].

2.4.7 Distributed Honeypot

Since malware and attackers must target honeypots, they must occupy a considerable portion of the address space[83]. The distributed honeypot system aims to fix current flaws that most previous honeypot could not fill[83].

1. Honeytokens

A honeytoken's worth resides on its usage. Honeytokens are phony digital entities with numerous uses. Honeytokens are fraudulent medical documents, faulty credit card information, and invalid social security numbers[83].

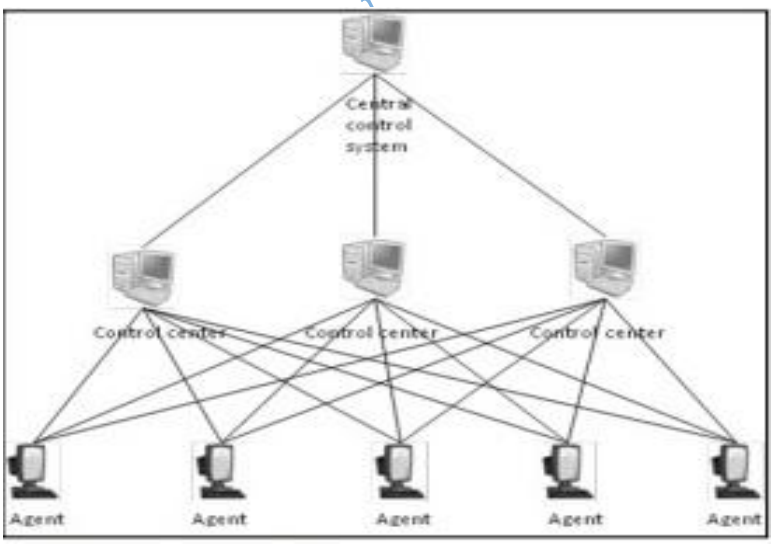


Figure 2.20: Distributed Honeypot[83]

2.4.8 Generations of Honeypot

"The Cuckoo's Egg" and "An Evening with Breford" introduced honeypots in 1991. Clifford Stoll's "The Cuckoo's Egg" was about a company hacker[84]. An author talked on "An Evening with Breford" which is about a computer hacker who is apprehended by Researcher colleagues. Honeypots were introduced in these two studies[84]. 1997's Deceptive Toolkit was the first honeypot. This equipment was used to counterattack. Honeynet was formed in 1999. The non-profit researched blackhats and shared their findings. Honeynet project created three Honeypot generations:

2.4.8.1 Gen I Honeypot

1999's version. These honeypots were straightforward to construct and deploy. Their data management and capturing procedures were minimal. Figure 2.21 depicts Gen I honeypots. Reverse firewalls regulate data in GenI honeypots and collect Intrusion Detection System (IDS) data. First, the firewall collects network traffic. IDS scans network traffic for suspicious behaviour and warns the honeypot administrator[84].

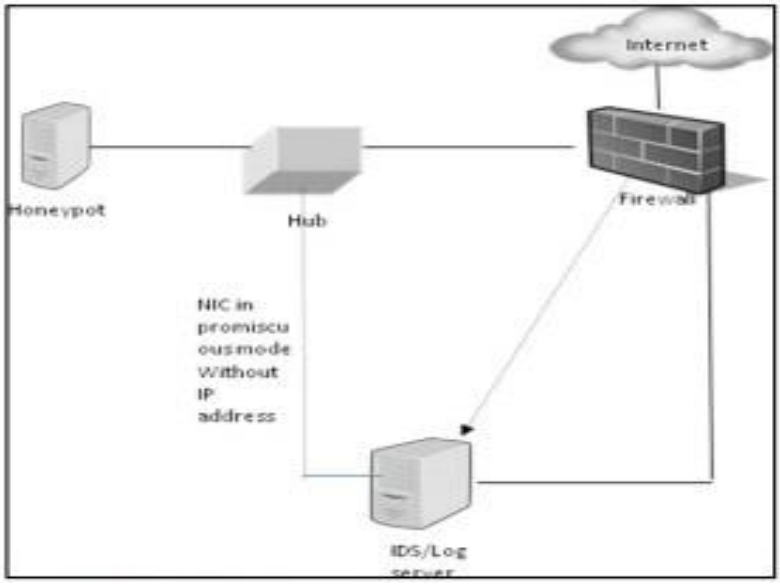


Figure 2.21: Generation I Honeypot[84]

2.4.8.2 Gen II Honeypot

2002's Gen II honeypots. Honeynet Project improved GenI honeypots after their success. Gen II honeypots aim for increased user involvement. This amount of contact increases risk, hence better data management and capture approaches are needed[84]. Gen II honeypots are in Figure 2.22 which shows that Honeypot hackers may target distant systems with the use of Firewall-based Gen II honeypots for data collection, analysis, and alerting[84].

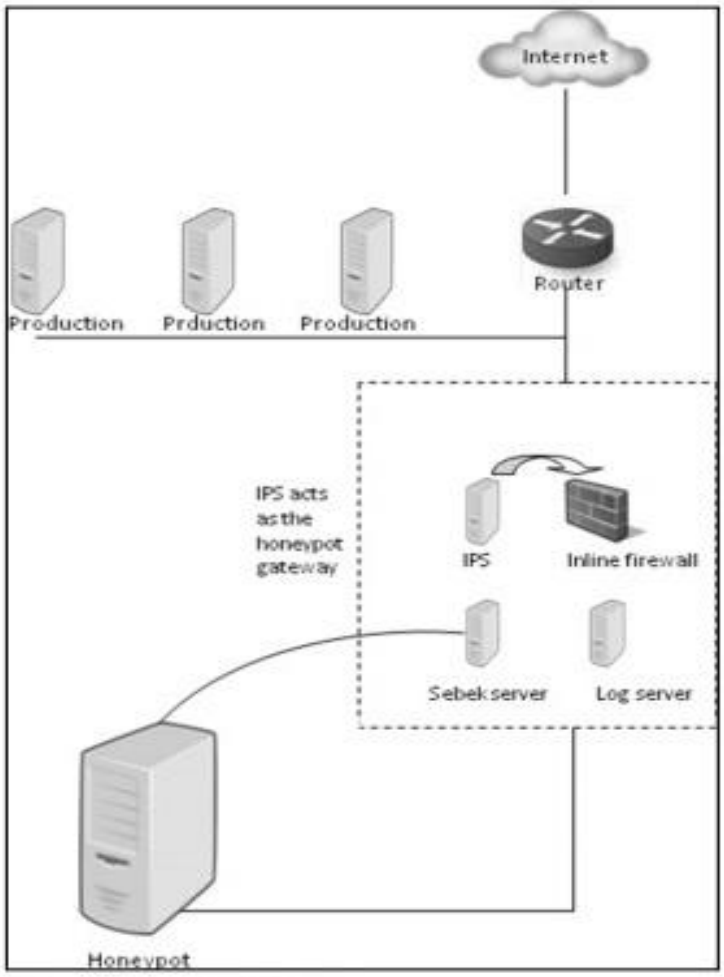


Figure 2.22: Generation II Honeypot[84]

2.4.8.3 Gen III Honeypot

In the year 2004 that ended with Gen III Honeypots, Gen II Honeynets fix Gen I flaws, while Gen II and Gen III honeypots are identical, however Gen III Honeypots with the Honeywall have improved deployment and administration[84]. Honeywall's interfaces allows two connections to connect the internal and external Honeypot networks. The third interface is for administration and setup and helps Gen III honeypots improvement:

1. Install, operate, and customize better.
2. New database structure improves data capturing.
3. Added a web-based data-analysis tool.
4. Improve UI and online documentation. Gen-III honeypot.

2.4.9 Tools for Honeypots

There are several Honeypot tools available. The three categories are data collection, analysis, and visualization. Data is collected via the Brazilian Distributed Honeypots project and the Matrix Chinese Distributed Honeynet deployment using CNCERT/CC[85]. Some works that are comparable to a log management server and a log analyzer include:

1. Honeyd

Niels Provos' open-source tool Honeyd manages virtual hosts on a network[85]. These virtual hosts may replicate several server types and network settings.

Advantages

• Ports and networks are monitored.

• It's free as an OpenSource solution and will expand swiftly with member involvement.

• Simulating operating systems at the IP stack and application level prevents fingerprinting.

Disadvantages

• Low-interaction means hackers can't hack it; and it lacks an integrated technique for informing and recording complete information.

2. Honeyview

Honeyview can summarize Honeyd log files in both text and visual formats. Honeyd is a great tool for getting info from hackers and script kiddies, but it may be confusing. Honeyview helps you save time while reviewing honeypot activity[85].

3. Honeydsum

Honeydsum is a Honeynet Perl log analyzer from Brazil. It may summarize Honeyd records IPs, protocols, ports, and networks[132]. It shows source, port, and hourly connections. It is capable of combining log files and honeypot events. Their summaries, on the other hand, are text-only[85].

4. DTK (Deception Tool Kit)

One of the first honeypots was Fred Cohen's DTK. It offers defenders a significant advantage over attackers. Honeypot with low-to-medium involvement. This is not a novel idea. Deception is used in counterattacks. DTK deceives attackers into thinking the system has several recognized flaws[85].

5. BOF (Back Officer Friendly)

BOF detects computer Back Officer scanning. The honeypot was not intended. It was a threat-specific tool. In response to the Dead Cow Cult, Marcus Ranum and Network Flight Recorder produced it in 1998. It can now identify connections through telnet, ftp, smtp, pop3, imap2 BOF mimics answers to hackers when it connects to one of these services, giving you time to stop them[85].

6. Specter

Windows honeypot SPECTER is low-interaction that simulates machine. It can imitate 13 OSes, monitor 14 TCP ports, and provide Internet services[85].

7. Mantrap

Recourse Technologies designs, maintains, and sells ManTrap. It's unusual because it gives hackers whole operating systems to compromise, logging every access. It's meant to be assaulted and broken. ManTrap creates a virtual minefield for internal attackers to traverse[85]. One step in the wrong direction and the attacker is exposed. Like the homemade jailed environment ManTrap creates a highly controlled operating environment that an attacker can interact with. ManTrap creates cages that are mirror images of the master operating system[85].

8. HIHAT

The High Interaction Honeypot Analysis Toolkit (HIHAT) allows to transform random PHP applications into web-based high interaction Honeypots. In addition, a graphical user interface is provided which supports the process of monitoring the Honeypot and analyzing the collected data[85].

The table showning the comparison of Various Honeypot Tools can be found in Appendix IX

2.4.10 Different Models Based on Honeypot System

There are different models based on honeypot system that have been researched and identified by different researchers[86]. These models are designed based on the purpose to which it is meant for and how it wants to be implemented. Below are the various models of

honeypot system according the purposes and level of implementation for attackers' sensitive information gathering[86].

a) Secure VoIP Architecture Based on Honeypot

b) Secure SSH Architecture Based Honeypot

c) Secure Phishing Architecture Based Honeypot

d) Secure Social Engineering Architecture Based Honeypot

2.4.10.1 SIP Function

A researcher defines SIP as a text-based protocol governed by the Internet Engineering Task Force[86]. This protocol starts, changes, and ends two-party multimedia sessions[20]. The following message types are supported by this protocol:

1. Options

2. Invite

3. Register

2.4.10.2 Attack Stages Related to VoIP Network

With respect to the SIP component, there are have three different stages of attacks and they are as follows;

1 Devices Based on SIP and Server Scanning: For a SIP-based attack, a network destination is required. Locate the target system's network[87]. OPTIONS and INVITE are sent to SIP devices.

2 Collecting initial information: After finding the target, the attacker gathers device information such IP address, calling Id, etc. Attackers may use OPTIONS and REGISTER to learn about authorized clients and services[87].

3 Client-attack: The attacker gathers remaining information, finalizes strategy, and launches assault. REGISTER message type with fake IDS builds a list of active users and transmits voice spam to the client[87].

• Honeypots should seem like actual systems to trick attackers.

• Determine whether a low- or high-interaction honeypot is needed for deployment.

• Honeypots should include tools for studying, storing, and verifying data.

• Have a healing approach to reinstall honeypot's invariable location.

Honeypot has three deployment positions. Based on honeypot's objective and operation, the top three sites are:

1. External Position

Honeypot has direct internet connectivity here. They're not separated. Due to these qualities, honeypots are more exposed and may be harmed by attackers, but they can identify undesired actions. Honeypots employ this form of positioning[87].

2. Internal Placement

In this configuration, a honeypot is inside the network and a firewall is outside. This site is used to detect internal hazardous system faults[87].

3. DMZ

Honeypot lies next to actual server DMZ to discover threats there.

2.4.11 Designing of Architecture

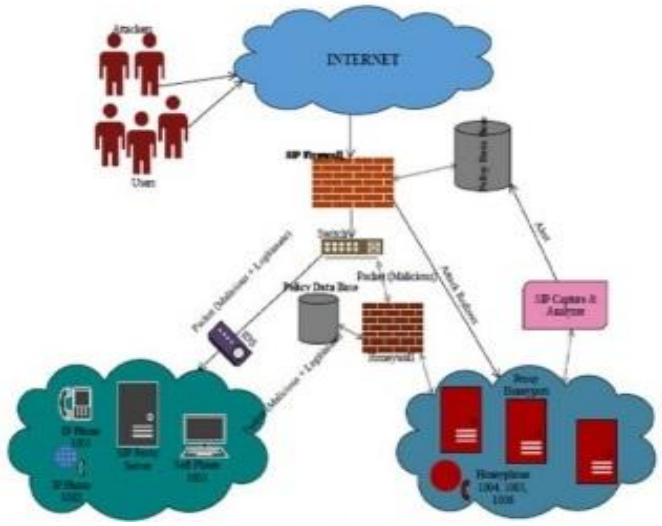


Figure 2.23: Secure VoIP Architecture Based on Honeypot[87]

The graphic demonstrates Secure VoIP architecture based on Honeypot. The architecture's key components are as follows:

1. VoIP Honeypot

This component uses SIP users, SIP proxy server, and SIP end for placement. This component's true network location is next to a firewall. This part promotes user interaction. This operates like a genuine network system, making it hard for an attacker to manage the honeypot[87].

2. SIP Firewall

This firewall protects an internal network. This firewall lets packets in/out depending on header information and database rules[87].

3. Honeywall

Honeywall is part of this architecture. The honeywall protects the honeypot. It helps set up and manage honeypot connections[87].

4. SIP Storage

This component records input, output, and reaction. This component uses database rules to discover attacks and exposes received data without alteration[87]. The defensive system is warned to cease further assault. Increasing missing data, capturing hacker info, gathering detection proofs, etc. were suggested[87].

2.4.12 Intrusion Detection Model Based on Honeypot Technology

The authors demonstrated how the effectiveness of an intrusion detection system may be increased by modifying it[88]. This concept combines honeypot knowledge, honeypot intrusion automation, and electronic forensics automation. The diagram below depicts an intrusion detection paradigm based on honeypot technology[88].

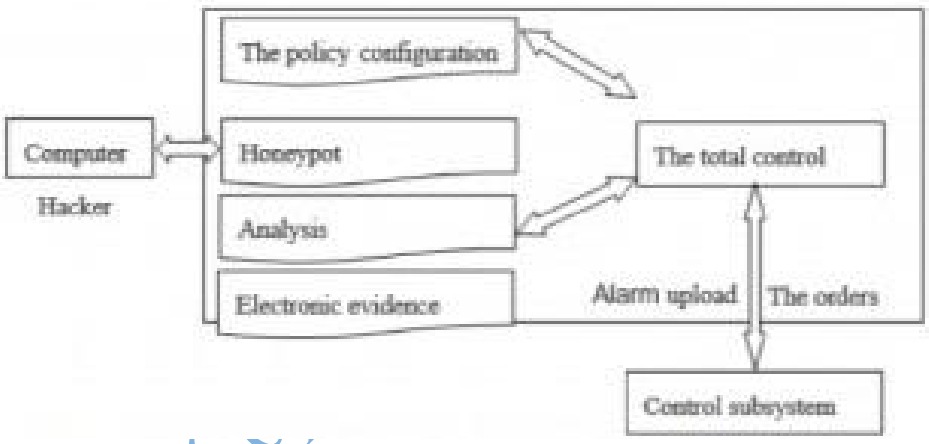


Figure 2.24: Intrusion Detection System Based on Honeypot Technology[88]

1) Policy Module Configuration

This setup needs opening or closing virtual machine ports and holes. Honeypots need data management, and storage.

2) Honeypot Module

This module is responsible for flow of in and out within a network which leads to any type of problem that must be recorded[88]. While coming to the network there could be some

problems like sniffer, malicious activities, and external network is shown to be breached etc. This module is used to handle all the incoming data and moving out of this model[88]. This module helps in keeping the eye on the hacker along with his/her malicious activities but also allow some of the vulnerabilities to the system which allow the attacker to be functional[88].

3) Real Time Analysis

Honeypot module works with real-time analysis module inside same network to detect disruption. This module employs Pattern Matching and Statistical Analysis for analysis. Statistical Analysis creates system behaviour statistics or a system template[88]. This template's selection and update are problematic. It can't analyze undesired behaviours.

4) Electronic Evidence

This module includes basic original data with intrusion effects to protect additional basic harmful operations. This module holds a large database of information that is compressed using multiple compression algorithms so that more data may be saved using less space, which is important for studying an attacker's nefarious activity[88].

5) The Total Control Module

Total control Module behave as the coordinator between different modules of this model and data which is interacting with this model[88]. This module along with the work of coordination also generates subsystem alarm in case of malicious activity and generates some command to every control unit of each module[88].

2.4.13 Automated Bot Control System Based on Honeypot

1. System Overview

The Bot Finder Honeypot (BFH) process is divided into two stages: training and investigation. Both levels function differently[89]. During training, statistical models were used to create malware family activities. To evaluate whether incoming data is from a dangerous device, extract features from statistical models and match units to compare data with malware family behaviours[89]. In Training, classification differentiates malware families. NetFlow runs samples once malware has been isolated. NetFlow collects data and routes it between IP addresses and to the target port. All malware vectors and their families are combined in a modelling unit. Using a clustering approach, all created feature vectors are checked for classification against each training-stage detection model in the matching unit. If this device generates an alert, it signifies that a trace's internal IP address is infected[89].

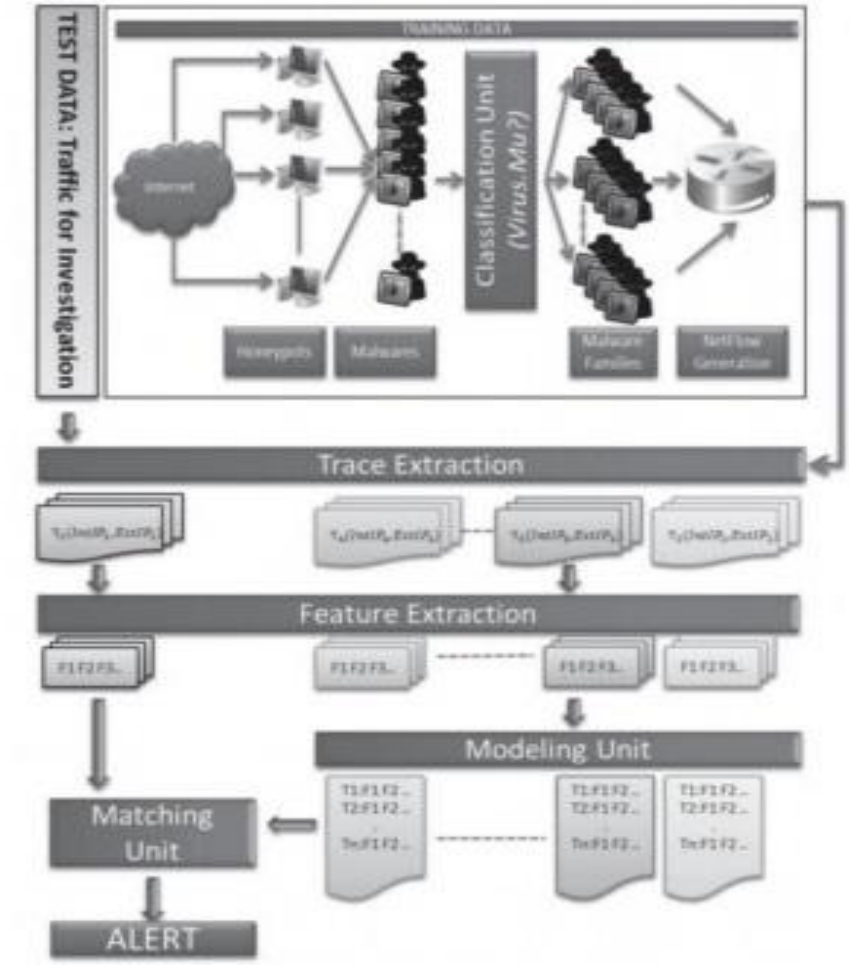


Figure 2.25 System Overview[89]

2. System Details

• Cyber Threat Monitoring System (CTMS)

This concept requires wide-area sensors. This model's data comes from Cyber Threat Monitoring System (CTMS)[89]. This system includes Distributed Sensors and Malware Detection Centre. Malware Detection Centre has submodules for honeypot, network traffic, etc. In this strategy, honeypots are used to classify incoming data[89].

• Classification Unit, Honeypots, NetFlow generation Honeypot

Honeypot has its specific features because of which it has its own importance in network security. It has its features like network traffic control, improving/handling network

services, operating system and applications[90]. Honeypots are defined based on their ability to handle malware activities i.e., low interaction or high interaction and also based on their roles on server and client side as well[90]. NetFlow: Malware families works in virtual environment and their behavior varies according to the situation, in the same way this model also alters its setting based on the changing original manuscript of malware activities like changing registration keys, contain VM keyboards, change MAC addresses based on Virtual Machine and also by finishing some of services of virtual machine which are related to malware activity[90].

It's like VirusTotal, which employs VM-related antivirus programs. If a file in queue is infected, it's tagged and a new naming scheme is started[90].

• Features Trace Extraction

First feature phase. Statistical models and algorithms extracted NetFlow data. Bot detection methods use traces to represent flow[90].

Feature Extractions: After extracting characteristics, the model uses ATI, average connection time, average source byte, average destination byte, communication regulation, and outgoing data accumulation regulation. Flow pair analyses all statistical characteristics[90].

• Model Creation and Detection Unit Model Creation

In a supervised learning algorithm, size and attribute are introduced first before introducing the training phase[90]. So, in this model, unsupervised learning algorithm is used i.e., CLUES (Clustering Based on Local Shrinking) algorithm, for creating detection design for every malware family[90]. In this phase, trace is analyzed for every six features of training data. CLUES algorithm is used foe clustering these features which gives rise to dynamic size

cluster without fixing number of clusters. A single cluster has huge number of trace features for a specific malware family, which is generated by classification unit[90].

Detection: Each investigation data set characteristic is examined with each harmful family detection model cluster. If T is in M, it's a hit. Hit score is then cluster-weighted. If this feature's total hit score is higher than its threshold, it will connect to model-M[90].

• Distributed Processing

This approach utilizes HDFS for large files. Name and data nodes are present. Data node stores and retrieves data, whereas name node handles file information. BFH models and matches IPinternal and IPexternal traces. Hadoop is a massive database with vast data thus the main problem is moving it over network, reading data, and writing it to disc. This form of database maximizes traces in one node and reduces network migration[90].

2.4.14 Honeypot Based Signature Generation Against Polymorphic Worm Attacks in Network

• Architecture

Honeytraps 1 and 2 are distinct high-interaction honeypots. Both honeypots have many levels. These are known as Research Honeypots. The major goal of installing two honeypot traps in a single system is to catch the attacker such that he/she is unaware of what is happening on and that all of his/her system actions may be recorded and appropriate action taken based on those activities. This solution uses a physical honeypot with three software layers: System, Sebeck, and Application. OS and Sebeck Client get main system protection[91].

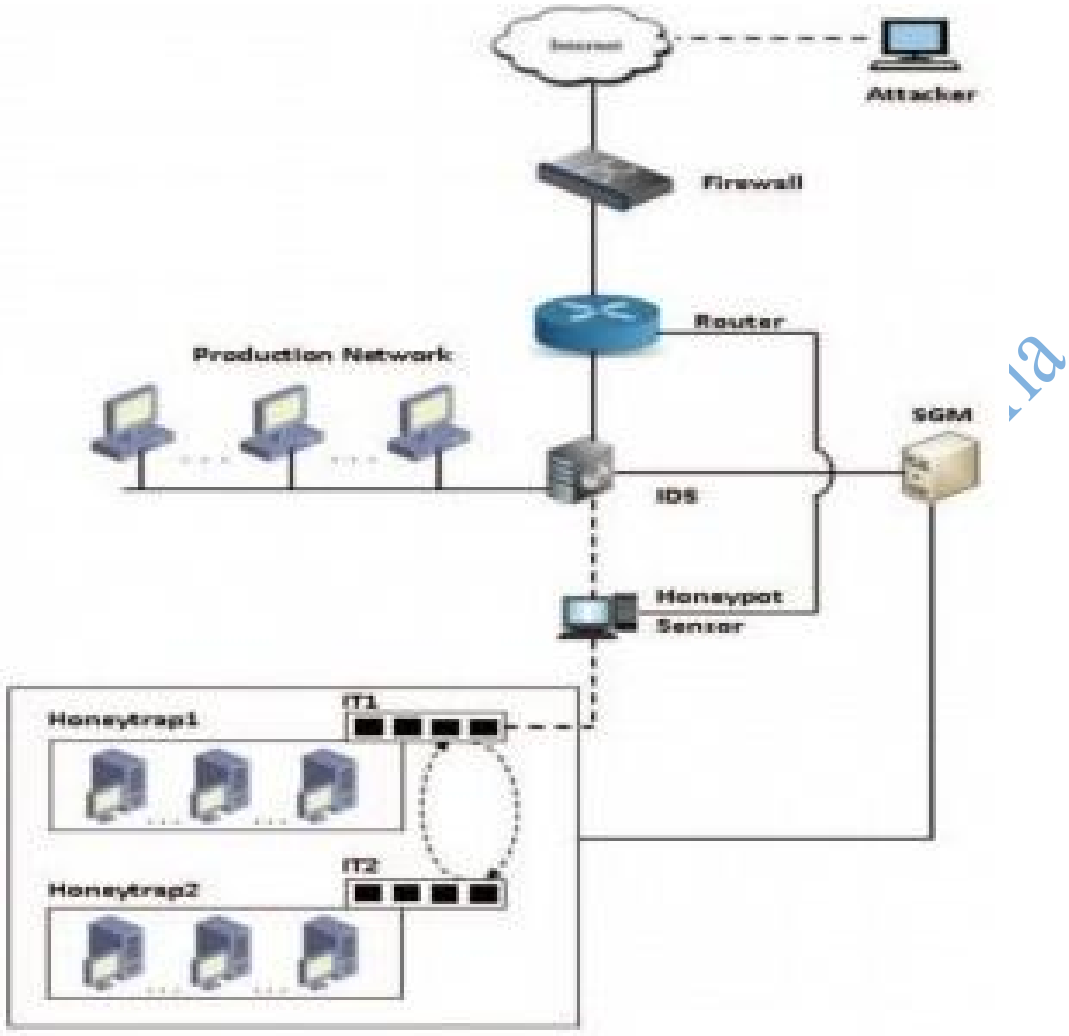


Figure 2.26 Architecture of Proposed Honeypot System[91]

• Data Control: Router-to-honeypot layer bridging isolates honeypot from remaining network. This bridge allows the attacker in, but he must leave[91]. Since bridge provides dual functionality in layer, an attacker cannot know IP address, MAC-address, routing path, etc. Honeytrap1 develops an outgoing connection to attack. The router's internal translator1 sends traffic to honeytrap2. Honeytrap2 tries to link to outside systems like honeytrap1. Honeypot2 transfers traffic to honeytrap1 using internal translator[136]. This system only enables certain connections since it serves two objectives, the system can store enough malicious traffic to limit DoS attacks[91].

• Data Capture: To analyze malicious traffic accurately and build a signature from it, several stages of information are needed, therefore a single layer is ineffective[136]. Firewall examines header information and blocks harmful traffic. Between router and honeypot sensor. Honeypot is the last step of this multilayer capturing system. It stores different kind of data required for generating signature from worms[91].

2.4.15 Classifications of Honeypots

A honeypot can be classified based on three categories namely, interaction level, deployment modes, deployment categories. These categories are further elaborated in detail in the following subsections.

2.4.15.1 Interaction Level

High-, medium-, and low-interaction honeypots: A high-level interaction honeypot replicates target network data[92]. In these honeypots, the information gathered about the hacker is enormous, but the threat is high since the attacker has access to all resources[92]. A medium-level interaction honeypot mimics the target network and provides resources and data.

Low-level interaction honeypots resemble the target network and give minimal resources and data. In this type of honeypots, the information gained about the hacker is modest, but so is the danger, as the attacker has limited resources[92].

2.4.15.2 Deployment Categories

Production honeypots and research honeypots are two deployment kinds. When a company's resources are threatened, it uses production honeypots. Threats are detected and

prevented. Network security businesses utilize research honeypots to detect and record hacker activity[92].

2.4.15.3 Deployment Modes

Honeypot deployment modes include deception, intimidation, and reconnaissance. In deception mode, a mimic network fools attacker. The attacker thinks the real-time network will guarantee he uses every hacking tool[92]. Interaction honeypot. In intimidation mode, the attacker is notified of system actions. A notice that the attacker's actions are being tracked drives away some, while the others are monitored and their information is recorded for future reference. In reconnaissance mode, hackers' tools and tactics are recorded and utilized to create IDSs, and to Monitor both internal and external assaults[92].

2.4.16 Applications of Honeypot

This article outlines some of the fields where honeypots are employed.

1. Unsafe Environment

Because honeypots are sensitive devices, they must be implemented in a secure environment to prevent unauthorized access to their IP address and port number. The use of a honeypot improves system security[93]. Honeypot provides an adequate step for improving efficiency rate of system relates to their security[93].

Figure 2.27 Unsafe Environment[93]

2. Protected Environment

In this, a firewall restricts access to the honeypot. This firewall protects the honeypot system's IP address and Port number from clients. This notion adds constraints without affecting continuity[93].

Figure 2.28 Protected Environment[93]

3. Network Security

Honeypot has network security applications. Honeypots protect an organization's network by luring attackers. By monitoring honeypot activity, viruses and worms may be found. Honeypots boost network security by detecting attackers[93].

2.4.17 Advantages and Disadvantages of Honeypot

2.4.17.1 Advantages

a) Data Collection

Honeypots don't need much data thus they save just valuable, restricted info. Reduces noise. Honeypot provides precise, understandable data.

b) Simplicity

Simple design and easy deployment make honeypots a popular approach in enterprises.

c) Resources

Many security technologies are overburdened due to network activity, whereas honeypots store only incoming data.

d) Honeypot lowers false positives and negatives.

e) Honeypots teach security experts.

f) Honeypot gives attackers fake data.

g) It provides admissible forensic evidence. If implemented appropriately and not marketed, it may be utilized as legal proof.

h) Honeypots may thwart intruders. Knowing a system captures and logs every activity may deter intruders.

i) A well-built and deployed Honey Pot collects IP address, attacker intentions, and attack activity.

j) Honeypots deflect intruders from the production system.

k) Honeypots are cheap. Downloadable versions are straightforward.

l) Honeypots may identify insider assaults by revealing insiders' tendencies.

2.4.17.1 Disadvantages

a) If an attacker doesn't transmit packets to the honeypot, it won't know about undesirable activities.

b) An uncontained Honeypot poses a danger to the network.

c) Honeypot requires time to fulfil its promises, hence proper administration takes time.

d) Honeypots can only monitor interaction. They lack perspective. They just perceive opposition.

e) Attackers may employ honeypots to harm other systems.

f) Honeypots are finger printable. An attacker can fingerprint a honeypot by its predicted traits or behaviours.

g) Honeypots, like other networking equipment, must be maintained.

2.4.18 Stealing Online Accounts

Cybercriminals use several tools to access internet accounts. Botnets, data breaches, account hijacking[94]. They're important to this theory, therefore there will be concentration on them.

2.4.18.1 Via Botnets

A botmaster controls a botnet of hijacked computers (bots). Legitimate system administrators and users are often oblivious. Botnets send spam and steal online banking passwords. Cybercriminals use botnets to perform DDoS assaults, like 2016's Mirai attack on Brian Krebs' site. Drive-by downloads and phishing or spam emails "enlist" PCs in botnets. Bots and C&C servers commonly communicate over IRC, HTTP, or P2P[69]. Social profiles and IoT gadgets may also be bots[94].

A social Bot masquerades as a genuine user in an OSN. OSN bots publish messages, upload material, and request connections. Socialbots are managed by botmasters, like conventional botnets[94]. Socialbots may steal personal data from unknowing victims' social networks (by scraping profile pages)[94]. The botmaster may use or sell this data for spamming, phishing, and identity theft. Socialbots reportedly swayed elections and spread bogus news. Socialbots, like conventional botnets, may do victims serious damage[94].

Infiltration and hijacking allow defenders to learn about and take over botnet communications, interrupting the cybercriminal operation(s) behind the botnet[94]. Both mitigation strategies are expensive and time-consuming against complex botnets, as they entail reverse engineering malware binaries and communication protocols. Botnets adapt to circumvent current countermeasures, making these jobs difficult[94].

2.4.18.2 Via Data Breaches

Cybercriminals also hack internet accounts by attacking insecure servers and terminals, resulting in large data breaches. SQL injection, password guessing, and social engineering assaults are strategies they use[95]. Recent large data breaches include Yahoo (3 billion compromised accounts), Adult Friend Finder (412.2 million), and eBay (145 million)[95]. Given the scope, severity, and regularity of data breaches, the security sector must discover enduring solutions[95]. This is the main impetus for this thesis's study on cybercriminals' use of stolen internet accounts. Since data breaches cannot be totally mitigated, the security community must understand what hackers do with stolen accounts to create better detection and mitigation solutions. Reusing passwords across internet services exacerbates data breaches. Strong passwords strain users, which causes usability concerns[95]. Users commonly use memorable but weak passwords to protect their accounts. Reusing passwords and using weak passwords makes it simple for thieves to hack various sites, even those without direct data breaches. Cryptographic hashing, password managers, multi-factor authentication, public-key authentication, and proximity authentication are countermeasures[95].

2.4.18.3 Via Account Hijacking

Cybercriminals see internet accounts as significant resources. They hijack accounts to get their data. Webmail accounts, for example, become "hubs" that collect sensitive information including credit card data, password reset information, and government ID papers. An assault on a webmail account may lead to attacks on other related accounts[96]. Automated hijacking uses botnets (as stated before), whereas manual hijacking uses spear phishing attempts[96]. When manual hijacking assaults succeed, the cybercriminal assesses

the stolen accounts to evaluate their worth and decides what to do with them, generally selling the account credentials in an underground market or discarding them[96]. Hijacked accounts might send spam and phishing messages to abuse the victim's confidence. Known contacts are more likely to pass past spam filters. Manual hijacking is harder to detect than automated[96]. Manual hijacking is low-volume and hijackers imitate legitimate users. Automatic detection methods are difficult to configure to identify manual hijacking. Manual hijackers may generally avoid detection[96].

2.4.19 Android Architecture and Mobile Honeypots

2.4.19.1 Android Architecture

Android is an open-source app platform. DVM runs Java-based Android applications (Dalvik Virtual Machine). Apps use native libraries. Both dalvik and native programs run in the sandbox and may write private data[97].

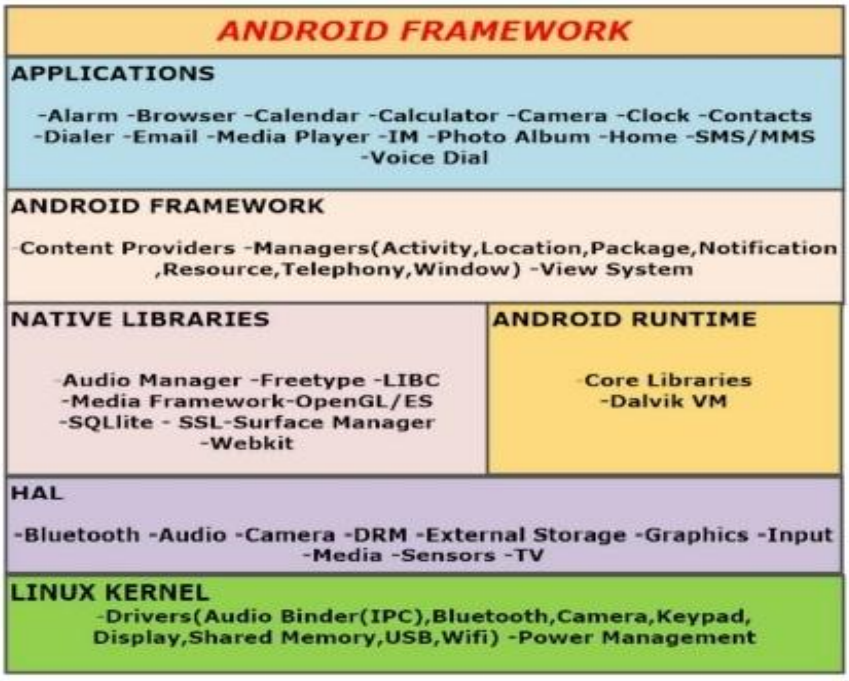


Figure 2.29 Android Software Stack[97]

Android apps enhance the core Android operating system, and the two major key sources for applications are:

• **Preinstalled Applications:** These applications have more permission as compared to user installed applications.

• **User-installed Applications:** This has less permission for third-party applications. Figure 2.29 shows Google's cloud services. Android isolates programs using Linux user IDs. It reduces an app's impact on other applications, the user, and the OS[140]. It prevents data theft. They can't use the network without permission[97]. Legally, Android repackages applications. Repackaging programs with added features may increase the number of malwares in different application marketplaces and increase profit and infection rate since the user sees the legitimate but the malicious is running behind. Android is Google's open-source OS. Android (Figure 2.30) APK contains numerous files. Android package includes executable.dex, AndroidManifest.xml, native code, and more (Android). Installing applications using AndroidManifest.xml[97]. This file contains hardware components, permissions, application components, and filtered intents.

Application-required hardware components are hardware components[97]. If a malicious program has network access, it may convey hardware information to an attacker. Benign software continually wants additional authorization[97]. This information regarding permissions is included in Manifest.xml. Content providers, activities, services, and broadcast receivers may all be retrieved from the file. Next is Intent, which represents communication between two programs. Malware may gather this information from the file or listen to particular intents[97].

Digital certificates authenticate the author itself to the application and Native code come as a binary file. The executable file consists of all functionality of the application and it can reuse the manifest file for the attack[97].

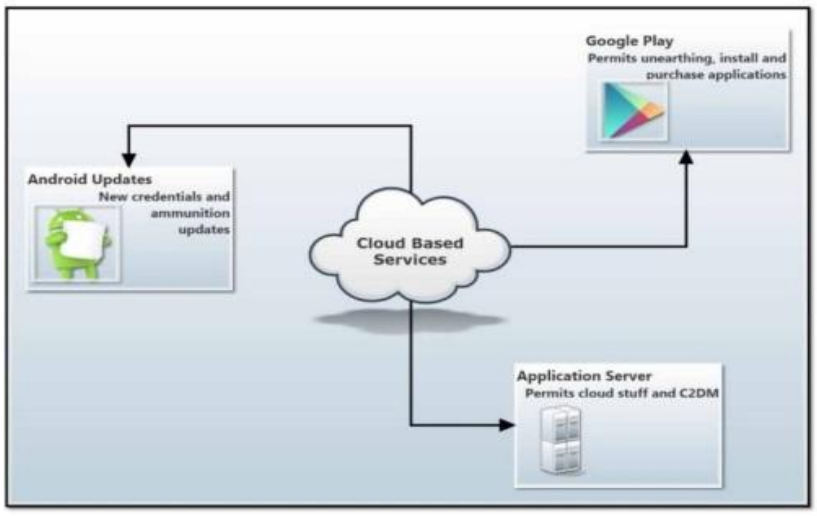


Figure 2.30 Services Provided by Google[97]

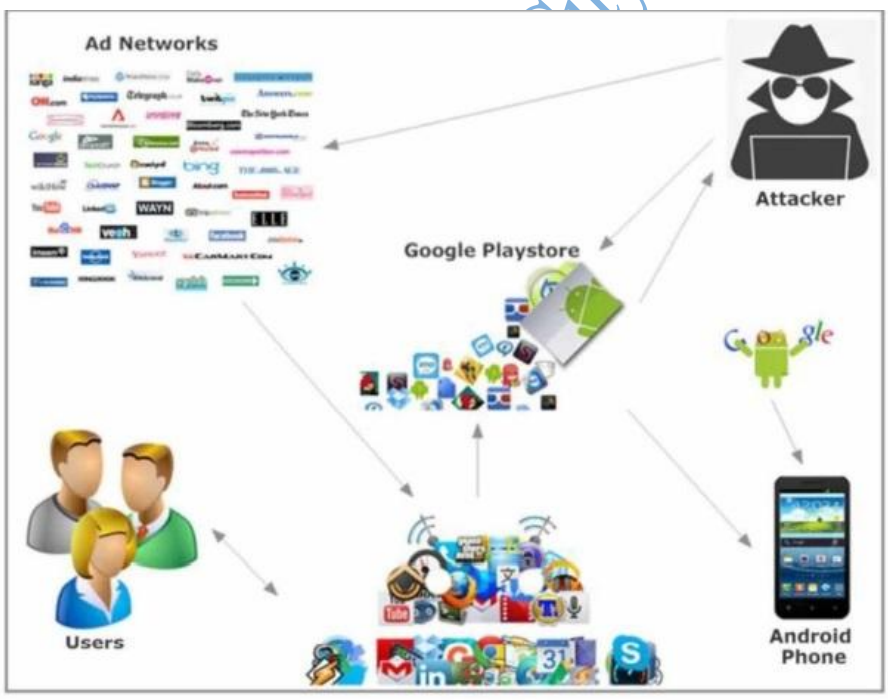


Figure 2.31 Android Ecosystem[97]

2.4.19.2 Mobile Honeypots

As cellphones become ubiquitous, assaults against them are prevalent, particularly because they are always linked to the Internet through WiFi, Bluetooth, and cellular networks. Many individuals use it to communicate. Operating systems must enforce smartphone security. Android dominates the market share. It's open source and Linux-based[97]. It's for touch-screen smartphones and tablets. Academic researchers suggest several ideas to enhance android security. These solutions depend on Android's kernel. Rooting a device compromises kernel integrity[97].

2.4.20 Challenges for Mobile Honeypots

A researcher explained Mobile Honeypot design and implementation issues as follows:

• **Visibility:** In order to attract attackers, honeypots must reproduce unprotected services at their ports. To minimize confusion, any honeypot application should respond to identified ports.

• **Monitoring:** The danger must be monitored without compromising the mobile device.

• **Audit Logging:** To repeat an attack, an uncompromised audit trail from monitoring is necessary.

• **Usability:** A mobile honeypot should be designed so that even inexperienced users may utilize it. This is made easier by the simple UI and setup.

• **Resource Utilization:** To avoid unnecessary processing overhead, the honeypot's power, bandwidth, and memory should be efficiently used.

2.4.21 Different Available Mobile Honeypots

Many mobile honeypots contain weaknesses due to Android's hardware and software vulnerabilities[98].

The following are some of the several Honeypots for mobile that have been developed:

1. Honeypot Labsac

Vladimir offered Honeypot Labsac to explore mobile device attacks through wireless networks.

They imitate services to test android attacks. It emulates telnet, http, and SMS. Connected mobile devices collect attack data. It may be used for different mobile OS and adding services[98].

Log Component analyses HTTP, Telnet, and SMS data. Telnet imitates itself to fool the attacker. False data via HTTP offers an attacker the impression of a legitimate server. Figure 2.32 shows data-capture situations[98].

This program emulates android services on many mobile OSes.

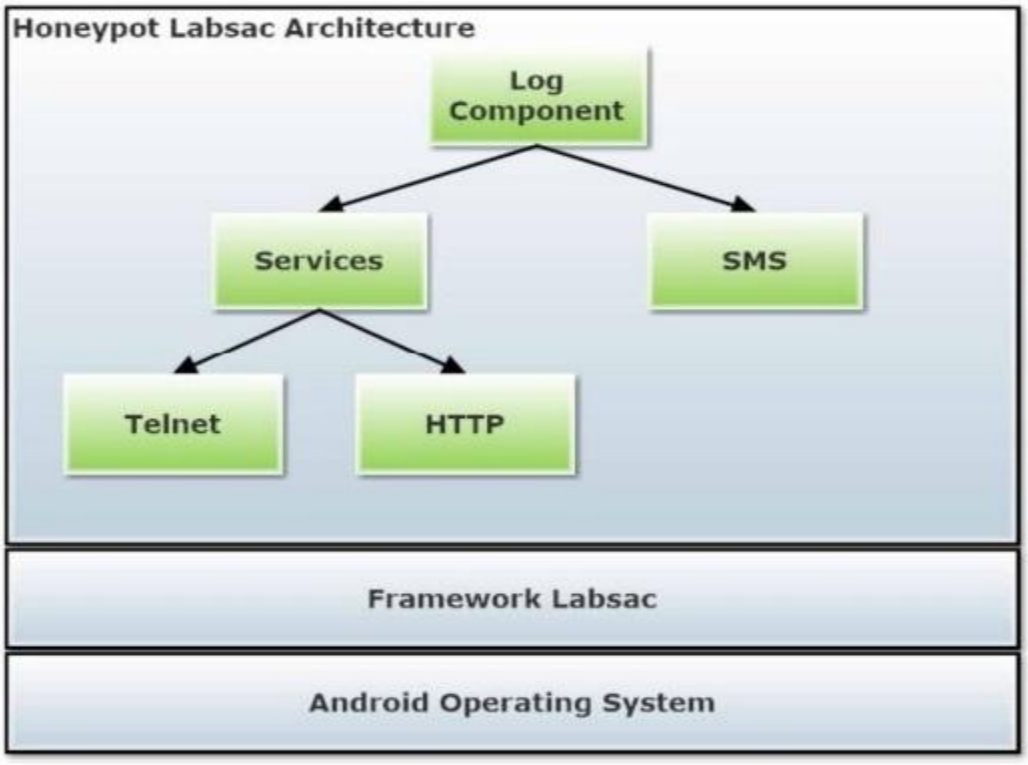


Figure 2.32 Architecture of Honeypot Labsac[98]

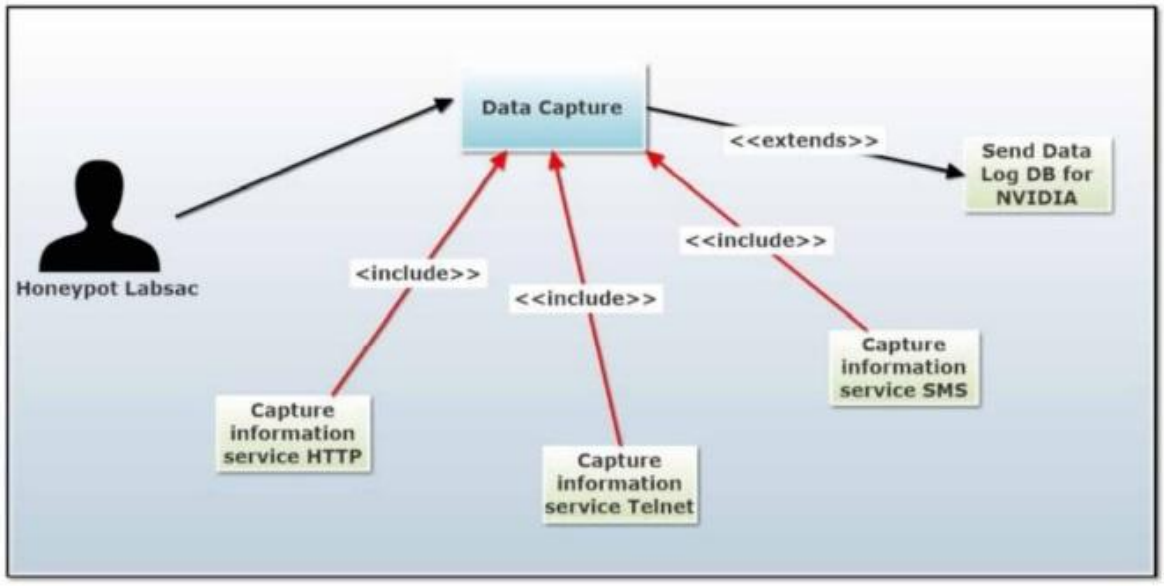


Figure 2.33 Use Case to Capture the Data[98]

2. Honeydroid

Manuel's writers designed HoneyDroid for smartphones. This creator works with hardware to increase its exposure. It's a smartphone honeypot using hardware-level virtualization instead of application-level emulation. The goal is to catch online, mobile, and malicious app attacks[90].

Figure 2.34 shows Honeydroid architecture. This honeypot runs L4 Android, a Linux OS that permits parallel computing. Binary-compatible with Android kernel, it builds instances. Microkernel isolates applications[99]. Log component stores information from other components with timestamps to preserve order. Android can't access log files, ensuring transparency and integrity. In this technique, independent hardware like WiFi and modem is virtualized, preventing android from directly accessing hardware and monitoring and controlling its interaction with hardware. Virtualization overhead prevents this honeypot from acting like an Android system. If hacked, microkernel's isolation will prevent an attack. Virtualization lets this honeypot repeat assaults[99]

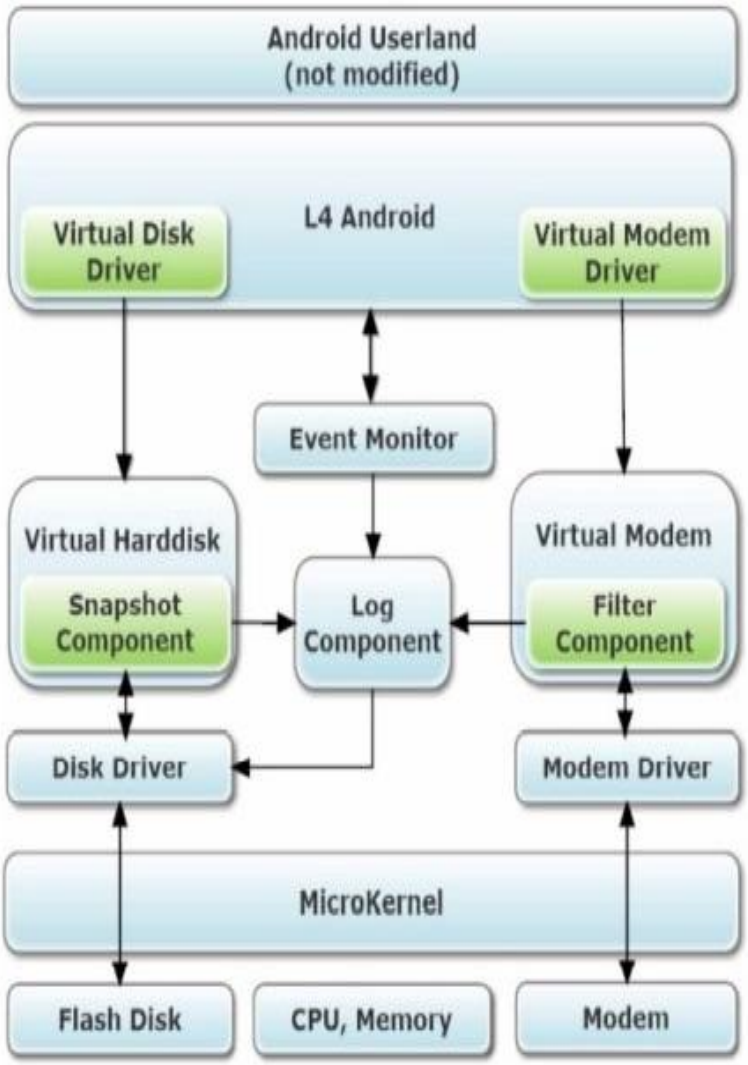


Figure 2.34 Architecture of Honeydroid[99]

3. Nomadic Honeypot

Liebergeld is a Commoner's Nomadic Honeypot. These authors offered SMS/MMS, NFC, malware, QR codes, and Bluetooth sensors. Smartphones are used in this honeypot. It is used to evaluate mobile applications, networks, and threats. The second virtual computer monitors and analyses the network[100]. Because the equipment has sensors and an operator-accessible safe back channel[100]. It can capture spyware from app shops. One itinerant honeypot has malicious WiFi and another has Bluetooth malware. Both gather and transmit

data to the operator, who informs other smartphone users. Figure 2.35 shows how Nomadic honeypot gathers data. Figure 2.35 depicts nomadic honeypot construction. Honeypot VM operates on mobile OS with its apps and infrastructure virtual machine. This infrastructure VM interfaces with virtual devices and honeypot VM to acquire threat data[100]. Even if compromised, this honeypot cannot be used to launch further attacks because all communication is done through infrastructure VM.

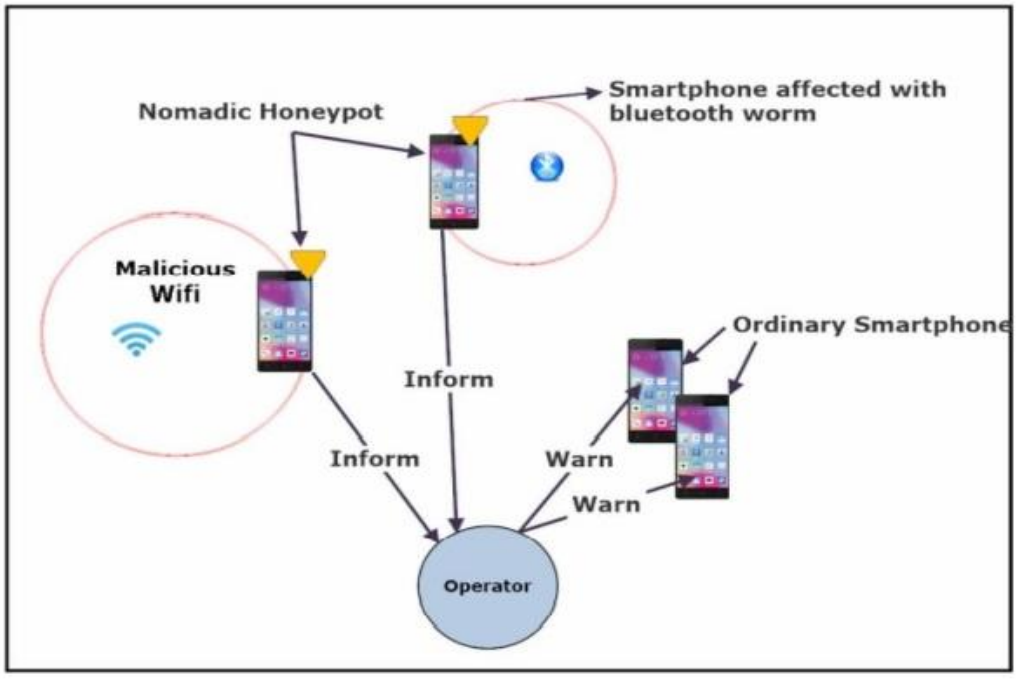


Figure 2.35: Concept of Nomadic Honeypot[100]

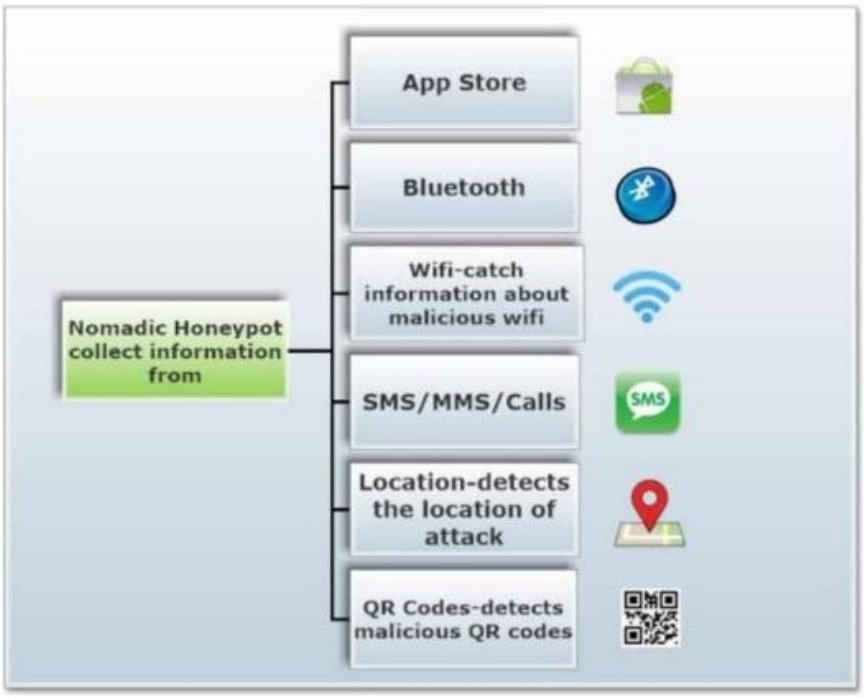


Figure 2.36 Malicious Information Collected from Nomadic Honeypot[100]

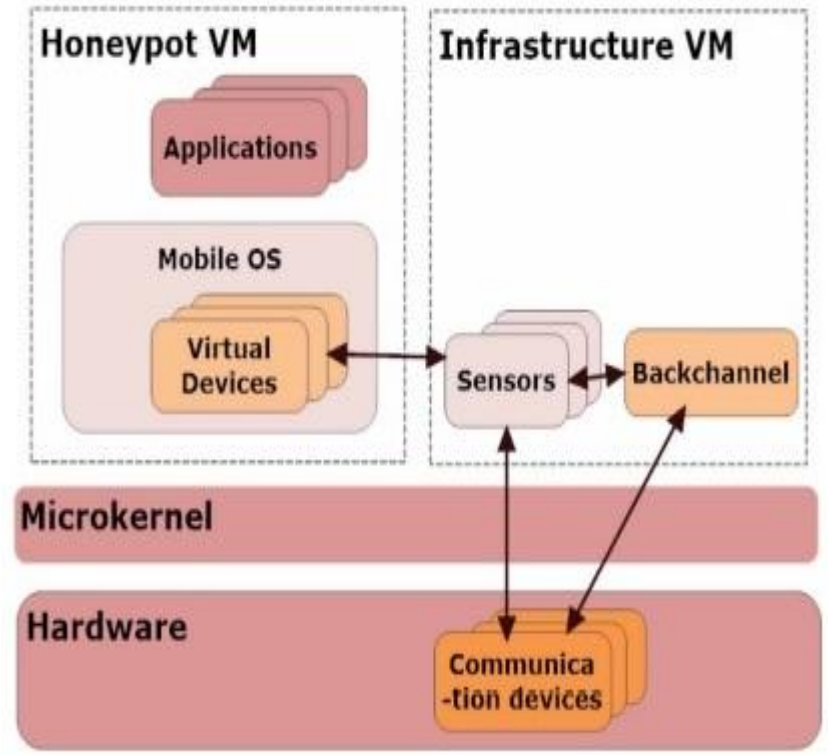


Figure 2.37 Architecture of Nomadic Honeypot[100]

4. Tracing for Honeypots

In Nikiforakis, the authors proposed Tracing, a cyber event monitor that obtains warnings from multiple continents by placing sensors. Analysis is done based on findings. Tracing, an open-source incident monitor, employs honeypots. Tracing shows the geographical location of the attacker and offers data[100]. Sensors are deployed throughout Asia, Europe, and America. It employs HTTPS for data and client-server architecture. Its sensors are:

• Dionaea Honeypots

• Mobile Honeypots

Honeypots may be readily spotted by using Nmap to search a network. Researchers should update default parameters and correctly handle port scanning. Tracing data over five months demonstrated that most attack-vulnerable protocols are old. Telnet and VOIP are attacked often. Large-scale port scans targeting several IP addresses were discovered[100].

5. Hostage Honeypots

Stephanie is a low-interaction honeypot. Its major goal is to quickly discover toxic networks and raise user security awareness. All new versions of android support Java for the app. Figure 2.38 depicts Hostage's assault surfaces and abilities[141]. DVM features a port binder, logger, hostage core, and GUI (GUI). Port binder connects sockets to ports and emulates user-specified protocols. Logger records intruder activities and alerts GUI. Figure 2.38 depicts Hostage's architecture[100].

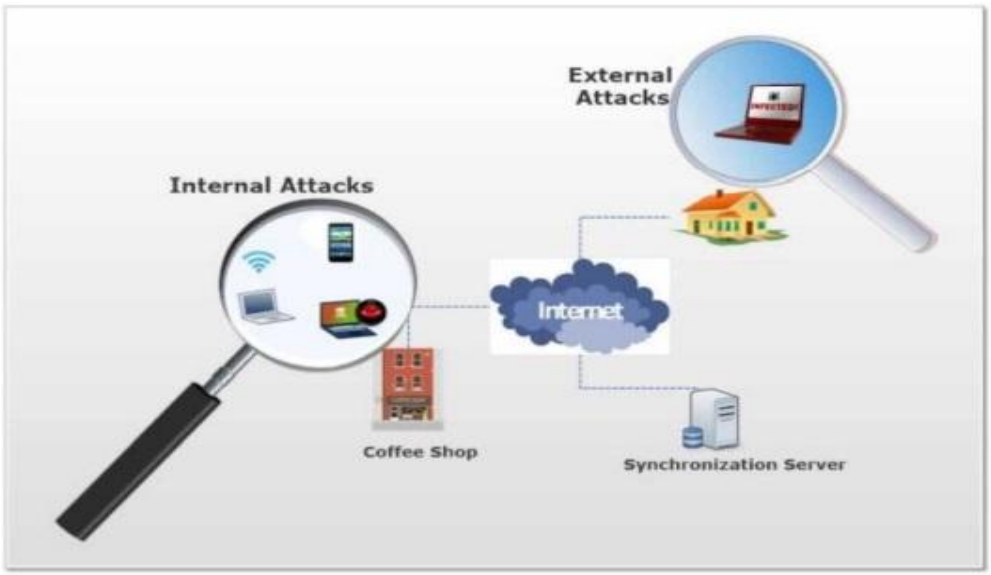


Figure 2.38 Concept of Hostage[100]

2.4.22 Limitations of Hostage

1. Initially, this constraint is a prerequisite for a rooted Android device, which restricts the user set's potential[100]. This disadvantage is resolved via alert data synchronization, which enables users to benefit from other Hostage-based devices. Consequently, the non-rooted smartphone may continue to be utilized by providing users with vital information through the ThreatMap function. This is a problem from a security standpoint.

2. Work is to be done on a graphical interface of honeypots, power utilization, and generic performance;

3. The functionality of learning about the security status of different networks from participating device exhibits a large number of challenges.

4. Hostage has malware peculiar action prior to detecting prevailing malware propagation that affects the time it needs to be agile in a network;

5. It is necessary to add a phone home capability to Hostage for asserting coarse grained algos.

As noted, before, mobile honeypots have distinct methodologies and features.

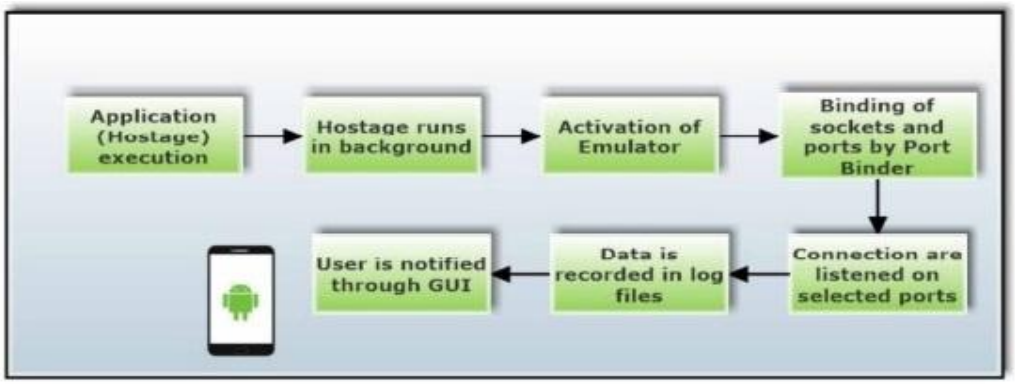


Figure 2.39: Working of Hostage[100]

2.5 Legal Issues in Cyber Security

Experts in cybersecurity must possess the same knowledge as hackers, namely black hat hackers. Unlike hackers, cybersecurity experts must adhere to legal constraints[109].

2.5.1 Personal Legal Issues

Cybersecurity regulations apply to anybody, not just employees. One may be able to hack a computer or network in the spare time. "Just because they can doesn't mean they should" Remember. Hackers leave footprints, whether they realize it or not[109].

Cybersecurity experts may utilize their expertise for good or harm. Legal professionals who defend infrastructure, networks, and privacy are in demand[109].

2.5.2 Corporate Legal Issues

Many countries have cyberlaws that guides their citizens which every individual should comply with. Infrastructures, networks, and the company and personal privacy may be compromised[109].

If anyone violates workplace cybersecurity standards, the employer may be penalized and such person may lose the job with possibly prosecution, fines, and punishment[109].

When in doubt, assume an action or behaviour is illegal and refrain from engaging in it. The organization may have a legal or human resources department that may answer one's questions[109].

2.5.3 International Law and Cybersecurity

Cybersecurity legislation is quite recent. Most nations have laws, and more will come[110].

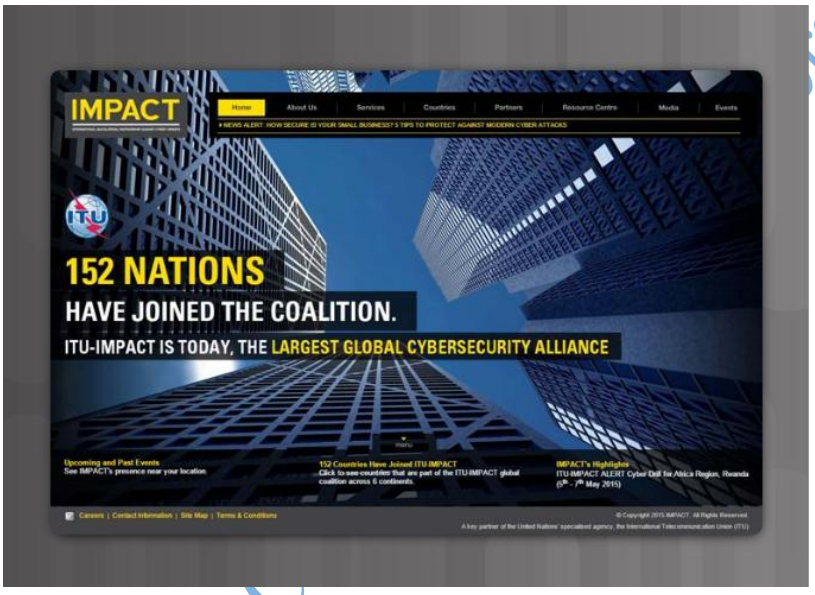


Figure 2.40: Ethical Issues in Cybersecurity[110]

In addition to working within the confines of the law, cybersecurity professionals must also demonstrate ethical behavior.

2.5.3.1 Personal Ethical Issues

Unethical behaviour isn't always prosecuted, fined, or jailed. Because it wasn't technically unlawful, the conduct is still unacceptable, However, ethical action is straightforward while unethical cybersecurity actions are tough to list[110]. Two examples are as follow.

• Would I want someone to hijack my computer and modify my social media images?

• Would I want to learn that a trustworthy IT specialist who fixed my network shared personal information with colleagues?If you answered no to any of them, don't do it to others.

2.5.3.2 Corporate Ethical Issues

Laws may impose ethics rules. Many cybersecurity aspects aren't regulated. Legal actions may not be ethical[110]. Many elements of cybersecurity are not (or not yet) addressed by regulations thus IT companies have formed codes of conduct. Three organizations have ethics codes:

• CyberSecurity Institute (CSI)

• Information Systems Security Association (ISSA) (AITP)

. Association of Information Technology Professionals (AITP)

2.5.4 Legal/Ethical Matters on Honeypots

Repudiation past articles on honeypots aren't lawful. Different authors acknowledge that U.S. laws apply to users. Consult a lawyer before deploying honeypots. Network management is concerned about the legal and ethical consequences of honeypots and their data. Networkers can't agree on acceptable activities. Honeypots monitor and collect user information, compromising attackers' privacy. Honeypots lose their function if monitoring is disallowed to protect privacy. Legal and ethical issues impact honeypots[110]. Recent conversations on this subject reached a quasi-consensus, although numerous worries remain. This part concentrates on legal and ethical issues with honeypots and reaches an agreement[110].

Rubin said Every researcher and security expert must ensure honeypots are lawful and ethical. Wiretapping rules restrict network sniffing, however there are exceptions: protecting rights and property, notifying the user, and government direction[110]. Frame-up may prevent a hacker's trial, but it must be demonstrated that the honeypot owner went above and beyond typical methods to encourage a hacker to do a crime he wouldn't have otherwise done. Attacking attackers is forbidden if the attacked system belongs to innocents[110]. The owner may be liable if honeypots are utilized for crimes. Honeypot owners should be cautious while studying viruses and worms to avoid unintended distribution. The user must also disclose vulnerabilities to a company long enough to implement patches before disclosing them[110].

This study investigated ethical and legal difficulties with honeypots. Honeypots have no legal precedent, thus most of the debate is well-informed speculation about the legality. Police use honeypots. When a system administrator sues attackers, safety and legal advice are needed[110]. Privacy standards do not apply to improperly accessed data, although gathering honeypot data may be deemed an illegal wiretap. Honeypots aren't controlled by common carrier legislation since they don't give user accounts. Honeypot users should seek legal advice concerning liability. A honeypot owner might be prosecuted if a hijacked honeypot is used to target other honeypots[110].

2.5.5 Anti-Detection in Honeypot

Honeypots track assault activity and plans. This prevents future attacks and gathers forensic evidence for prosecution. Honeypots observe intruders without their noticing. Honeypot detection is vital for attackers thus they've constructed numerous customized honeypot detection techniques. Honeypot detection and anti-detection is a hot topic[110].

The Author studied Honeypot Hunter by Send-Safe. Honeypot Hunter built a local mail server and connected to a target system proxy. This prompted target-to-self proxying. Honeypot Hunter can identify a honeypot if the target says the attempt was successful but the mail server can't see the connection. Honeypot Hunter sends and checks delivery of test emails. Except honeypots[111].

The author researched advanced botnet design in honeypots to avoid detection. Wang created a peer-to-peer botnet that's difficult to shut down, monitor, and hijack. The hybrid botnet needs two types (Servers and Clients). The servers are publicly accessible, but clients hide behind private addresses or a firewall[111]. To prevent specific bots from becoming global, the planned botnet is hard to shut down, even when certain bots are detected and their peer lists are collected. The authors recommended establishing several honeypots with static IPs to douse peer-lists and shut down a botnet. Many fake bots may be required[111].

The researcher correlated four IP address shuffling methods: UBS (Uniform Block Shuffling), PSS (Per Source Shuffling), SGS (Source Group Shuffling), and NBS (Non-Uniform Block Shuffling), performing mathematical analyses on shuffling interval, data structure size to keep track of address blocks for honeypots and production servers, number of live connections likely, and resilience against attacks. Experiments comparing packet delay, packet-loss rate, and connection disruption rate showed no significant differences[111].

2.5.6 Honeypot System Configurations

Tuning honeypots is another obstacle to their widespread usage. Honeypots watch attackers' actions without being noticed. Operating Honeypots should be set carefully to

attract the right targets. Honeypot users and designers must determine what roles to monitor, whose to watch, when to monitor, etc[112]. Errors in setting make it hard for honeypots to attract prey and allow attackers to hijack them. Instinctively and erroneously collecting network activity will complicate data processing. Honeypots must be set for specific goals. This section explains how honeypot research simplifies and automates honeypot designs to increase their effectiveness for certain goals[112].

Researcher used honeypots and honey tokens to detect domestic threats. Unlike outsider assaults, attackers inside a domestic domain have easier access to the system since they are more acquainted with it. Honeypots should be deployed into the network to identify malicious activity inside the internal domain. As they know the system, honeypots must be very interactive[134]. Instead of expecting insiders to uncover honeypots, assist them. Honeypots must provide attackers information they desire but don't need. Examples: fake business proposals and design requirements. Honey tokens contain fake documents and server logins[112].

The Honeyd Configuration Manager is used to install minimum interaction honeypots. The manager utilized Nmap to search for OSes and open ports. After network scanning, the configuration manager installs low-interaction honeypots. Managers may configure each honeypot's exposed ports, network address, and simulated server activities. Passive network scanning to establish a honeynet, dynamic modification of operational honeypots, and overall system emulation enhancements[112].

Another author dynamically allocated honeypots. Automatic honeypot circulation is suggested. Medium and high interaction honeypots trap and analyze unknown attacks.

Formulas improve honeypot transmissions[136]. Intranet, firewall, subnetwork, and DMZ zones were created. Once a network assault is identified, the system determines honeypot zones, level, and quantity. The testing showed that the machine could reduce and prevent assaults more than static installations[112].

2.5.7 Honeypots, DDoS

This section addresses Honeypot deployment to stop DDoS attacks in hostile environments including the Cloud. In another development, the author recommends employing a Honeypot to combat DDOS assaults. Even a successful assault on a prey system may save other systems but not a honeypot[112].

Researcher exhibited Mirage Honeypot's educational and productive powers. Mirage converts Snort into a Honeypot. Compromise, impenetrable covering, and rapid analysis[112]. The author discussed Honeypot's fixed location and explained intruders can't predict roaming Honeypots' movements. External and internal DDoS assaults are blocked. This method involves choosing the right number of active and Honeypot servers.

The author created hybrid honeypots. It monitors manufacturing computer intrusion activity. Low-interaction honeypots mimic services and operating systems. It leads malicious users to a high-traffic site. Honeywall can be improved by installing it in every sub-network, actively analyzing invader behaviour, and without blocking legitimate users[112].

Researcher suggested modelling ISP DDoS in NS2, it detects, characterizes, and redirects honeypots. Entropy identifies low- and high-rate assaults. Further tiers track attack flows in real time and route illicit traffic to honeypot to learn about attackers while ensuring QoS for genuine users. By simulating real traffic, attackers may circumvent this detection

technology and launch DDoS assaults. Author added dynamic honeypot routing. Attack traffic is routed to a dynamic honeypot after detection and characterization. In 2008, the author creates active and dynamic honeypot servers in DMZ based on traffic load to prevent attackers from reaching the final target while balancing the load. Author then discusses the dynamic Honeypot[112].

Reseacher's attacker and unreachable problem are resolved. First authenticate client, then assign communication port. Later, one may be unraveled by building a virtual temporary communication channel using honeypot[112].

Another researcher used Honeypots and an ant colony optimization technique to counter DDoS assaults. Attack intrusion modules. Ant Colony algorithm updates database, directories, files, and other honeypot contents[112].

This author utilized prior methods to construct a DDoS architecture in 2009. Normal and naive decrease false positives and negatives. This auto-responsive architecture filters fake traffic and employs dynamic honeypots to discover attack services and new signatures. This strategy compromises response time for client throughput (CL)[112].

US and China attack HTTP and SSH first, research says. Amazon EC2 and Azure have huge attack volumes, although Azure mostly sees windows-based assaults. EC2 and Azure are under examination[113]. Low-interaction honeypots generate meaningless data. Unix-based systems make honeypots harder[113].

IP tracing was used to avoid flooding, because of this reason, most large, networks don't use this model. Data Centre logs are full[113]. Future research promotes distributed load-sharing.

These authors have described a defensive solution employing anomalous intrusion detection, Honeypot, and attribute-based access control to disclose real-time traces and discover new attack patterns (ABAC). Honeypot or ABAC are employed to defend the system, Venus Flytrap emulating HTTPS, SIP, FTP. This method revealed unlawful URL infections[113].

IHoneycol is an ISP incentive-based mitigation technique. Firecol-IPS and Honeypot-IPS block traffic close and distant from its source. It's helpful. This tactic often thwarts the dual assault and ping of death attack[113].

Honeypot technology was used to identify cloud intrusions, Brokers access cloudlet data using this approach. Honey gateways detect assaults on each cloud node tier. Implementing it at each level decreases this technique's processing speed, which intruders might exploit[113]. AmpPot imprints amplification attacks, this has made Researchers and intruders check public amplifiers to see how AmpPot identifies them. This approach used darknet and 21 Honeypots. Bots aren't only behind fingerprinting assaults, it added. It enhances vulnerable ports. It uses UDP port for multiple services, therefore it might be identified and utilized in an attack. Researcher used virtualization to create an attack information web service. HoneyVM machines can scale using the described technique. A backup server is constructed in case of a web service attack. Without hardware, it's cost-effective and efficient. HoneyVM augments made dynamically may be inefficient

Honeymesh mimics servers to monitor traffic and detect threats. It updates its routing information and blocks that source's traffic, detecting and combating DDoS assaults. Using many servers is costly[114].

A researcher detects unauthorized access points. They designed a hybrid method to reduce false positives[145]. System includes filtering engine, anomaly sensors, and shadow honeynet. Hunting second-level assaults and giving the attacker a simulated genuine environment improves performance[114]. Although Passive attacks that were detected were criticized with the use of Honeypot's fingerprinting. Due to poor data management, the author created Honeymix, an intelligent Honeynet with SDN. This method chooses the most desirable connection and checks for a honeypot. It centralizes service disruption and SDN switch policies[114].

A researcher talked about IaaS infrastructure combines three Honeypot technologies. Recommend high-security cloud infrastructure. Honeyd lures invaders. Honeywall removes intruders. The author creates signatures. Combining various IDS on a single cloud presents a diverse danger. Effective ways may boost efficiency.

Shuang use roaming virtual honeypots in an Ant-based DDoS detection system (ADTRVH). It tracks intruders using pheromones. It stores attacker information in multi-level IP log tables[114].

A researcher studied system data volume of the system[114]. Estimates rule-based mitigation strategies. Authors provided a way to get attack data. Their data-visualization technology generates mitigation rules, simulations, and evaluations. This filter mixes and separates packet properties which make the method isn't practicable for every scenario and parameter. Researcher created (HIDE), a proactive defensive system that mutates network host addresses and fingerprints[114]. This method has a substantial network overhead. Network stops operating between mutation algorithm changes[114].

2.5.8 Datasets and Available Tools

It includes datasets and scripts for testing mobile honeypots and cloud intrusion detection. It may also be utilized to stop DDoS assaults, so defensive strategies must be holy, robust, and trustworthy in order to detects unknown assaults[115].

Android malware research dataset records 5560 malware applications from 179 persons are found in this dataset. It analyses application code and manifest file statically. Android honeypots may utilize this dataset to guarantee reliability and security[115].

1) Malgenome: This contains android apps. Using numerous characteristics, it ranks android malware families from August 2010 to October 2011. 1200 viruses[148]. Android is split to reveal antivirus by passers. It lowers android malware;

Morpheus employs automatically-generated algorithms to identify android emulators. It analyses emulator and device sandboxes. Artifacts are then analyzed to construct rank detection algorithms[79]. This provides 10,000 scan criteria for emulation-based malware programmes[116].

2) KDD: Prosaic dataset describing an intrusive military Network entry. 4.9 million distinct protocols with well-defined source-to-destination phasing lengths. Normal or attack packets. Each category contains 23 values and 41 parameters[116].

3) Darpa includes two scenarios: 1.0, 2.0 LLDOS Former knows DDoS. Later, foreign attackers use DDoS[148].

4) Kippo-Graph (Brute force,): Results from Kippo-SSH honeypot can be conjured by jammed featured script, Kippo-Graph in the fashion of 24 charts that show 10 loftiest passwords, usernames, its combos, successful inputs, the ratio of success, inquest per week using Google's visualization technology[117].

Honeydrive (Brute force,) is a distributed honeypot OVA on Xubuntu 12.04.4 LTS. Preconfigured with Kippo, Dionea, Honeyd, Conpot SCADA/ICS. Honeypot, Thug, etc. Malware anatomy, forensics, and network monitoring[148]. It's in VirtualBox. • Omnet++ is a network simulator for testing honeypots performance and effectiveness and to optimize the trun around time of the execution. Eclipse supports Omnet++. It allows developing simulations, adding wizards, editors, and other features. It can simulate wireless and wired networks, protocols, queuing networks, multiprocessors, and other discrete characteristics has been made to serve as a commercial tool[117].

2.6 Summary of Gaps in Literature Reviewed

During this examination, it was discovered that a number of researchers have undertaken various work on cyber security deception technologies. Honeypot systems, in which attackers may be deceived and lured into the system using a fake application as bait in order to get sensitive and intelligent information about the attackers, were strongly emphasized. The majority of studies concentrated exclusively on gathering knowledge about the attacker, which has proven inadequate to secure a company's vital data. In this regard, our study has provided more insight into the actions of attackers in the area of developing a fake honeypot system to function in concert with the real honeypot system in order to also deceive the system administrator. This work has contributed to the body of knowledge on deception technology in the development of a honeypot system that may gather intelligence information from attackers and also detect a fake honeypot system that attackers may establish.

Honeypots are designed to fool attackers and may be used to identify illegal access to sensitive information or record attacker behaviour. In this part, there was description research that use deception-based tactics and honeypots[101]. Several literatures on Deception technology, Cyber security, and dangers were investigated for this suggested study to grasp past research and GAP areas[101]. The below-listed literatures discuss the creation and deployment of deception methods using decoys and honeypots as deception tools to advise against unjustified danger that might harm cloud data[101].

1. In a paper titled: Honeypot, a Security Tool in Intrusion Detection, thus discussion on the honeypot, which serves as advanced security tool minimizing the risks from attack on IT and networks[101]. They concluded that security experts should replace existing security systems with honeypots, but they did not anticipate that honey alone could not provide adequate security.

2. In "A comprehensive examination of honeypot strategies," Some honeypot researchers provide honeypot features and discovery avoidance methods that restrict honeypot identification rates[101]. The researcher developed a technique to detect new attacks in low-collaboration honeypot traffic. The suggested recognition method is two-staged[101]. First, IP-based traffic streams are constructed, followed by a PCA profile of honeypot traffic. The author primarily focused on low interaction honeypots, which are simulated and don't give as much security as high interaction honeypots[101].

3. In a study titled "A Survey on Honeypot Technology: Concepts, Types, and Working," Idea of honeypots and how they are meant to attract intruders so that their actions may be observed without endangering production systems or data[101]. Honeypot must be

periodically revised to include new techniques and assaults in order to guarantee protection against new types of attacks, as shown by the paper's findings. Honeypots may be used for either production or study[101]. Compared to other security techniques, honeypot configuration is straightforward. The author focused primarily on the capabilities of honeypot systems to acquire intelligence on attackers, but did not account for the possibility that attackers may establish false honeypot systems to confuse security managers as to which is the genuine and which is the fake.

4. In spite of different deception technology techniques that are being deployed by cyber security experts to prevent attackers from gaining access to the cloud information, the threat increases every second as attackers continuously exploring different means to gain unauthorize access the cyber space. Therefore, there is a need for effective, robust and more efficient models to mitigate threats poses by Cyber criminals. Hence the need for this research that implements a web based low interaction Glastopf honeypot for Accurate gathering of Attackers information and detection of fake honeypot.

This project addresses the capacity of attackers to create false honeypot systems to trick and confuse information security administrators[101].

5. Another Author wrote on "Improving Deceptive Honeynets via Empirical Learning". This study covers honeypot technology and explains ways to increase deceptive honeynets' efficacy[101]. The paper's results show that after testing systems with attacking tools and improving them based on pre-selected hacker attacks, systems would be further tested to determine if the level of deception has improved[101]. The learning approach that was

adopted for testing could only test but lacks better features for correlating results with existing honeypots, which could affect its level of performance determination[101].

6. In the Tularosa Study, it was explained that deception impacts on an attacker's reconnaissance and exploitation[101]. Cyber deception affects attacker performance by increasing time spent on decoys and self-reported perplexity. The author thinks only deception technology can affect attacks, although other security techniques may increase honeypot system performance[101].

7. Deception was explained as a defense and compared APTs to insider attackers. A researcher disrupted attackers' (APTs') network scanning and another researcher presented a mechanism to automatically produce and inject false network traffic. Another author of honeypot used honeytokens to "bait and deceive" information-stealing malware[101]. HoneyGen produces genuine honeytokens based on actual token rules. HoneyGen needs a real-token database. Another researcher created a deception system based on fake papers and false credentials[101]. They specified a set of honeypot attributes (requirements).

8. A prototype honeypot system with tens of thousands of virtual honeypots was created. The activities of Secure Shell (SSH) attackers were analysed[101]. By studying attackers' ssh terminal instructions, they sought to discriminate between human and automated attacks. HoneyLab enables organizations to build honeypots on shared computer infrastructure[101]. Another author presented and explained vividly about the HoneyDroid as an Android-based honeypot. The author created a virtual web client honeypot that can analyze JavaScript and VBS[101].

One of the authors improved Linux sandboxes for malware investigation. Another researcher installed honeypot devices and studied attacker behaviour in one of the articles[101] Cybercriminals might use internet accounts to research their activities. Social spam in OSNs and email spam have been studied using online honeypots. DeBlasio analyzed hijacked websites utilizing honey webmail accounts. They tracked honey account accesses from data breach websites. Attackers used password reuse across internet services. Other research examined offenders' webmail and cloud document honeypot behaviour[101].

9. As these studies and others in this chapter illustrate, most research focuses on identifying malicious accesses. An experiment to quantify cyber deception was examined. Honeypots fool, contain, and observe invaders, as it was proved. This method was improved. Technologies fool an enemy's network topology. Summers created a framework for deception to better analyze human, computer, network, or human-computer deceptions. Spitzner Cyber security simulations illustrate how various honeypot designs might influence attacker behaviour[102]. Fake honeypots were devised after real ones deterred attacks. Honeypots and decoys are deceptive. Decoy systems are low-fidelity and integrated into the real network. An attacker's network topology is hidden through decoys. Provos said decoy systems' usefulness hasn't been proven. Few studies have been done, and none contained rigorous experimental controls or a big enough sample size[102].

10. Another author argued that neither participant pool is good for cyber attackers. Deception on internet-facing network nodes produces aggressive behaviour, but it's unmanaged, opaque, and no complaints, interviews[102].

This was also said by another author that students lack the knowledge and attitudes to fool expert foes. In this investigation, it will use the closest analogue to hostile cyber attackers available for scientific testing–red teams–and bring in a big enough number to give statistical power and reliability to reveal impacts[102].

This was corroborated by another author that advocated several varieties of honeypots in their research. Cyberdefense uses deceit. This portion explored cyber deception modelling and planning studies. Formal methods literature contains logical deception models. The author suggested multi-modal agent communications that provides a computerized deception planner that receives operating system commands[102].

11. A researcher gives formal deception planning. Game theory models attacker-defender deception. Signaling game is a deception-planning concept. Ettinger model credibility-based bargaining[103].

12. Another author explores deception's influence on attacker-defender collaboration. Game-theoretic models, although obvious, aren't communicative or accessible in a larger context. Another is probabilistic deception models. Rowe utilizes a decision tree to decide whether to perform a deception[103].

A probabilistic obstructive counter-planning model was proposed by an author in another work for probabilistic strategy for defeating network scanners with false scan results. Existing models aid with deception modelling and planning, but none is generic enough to design cyber deception against numerous threat models[103].

This section summarizes pertinent prior work from different researchers of honeypot deception technology. The company must secure cloud assets. Several leading public cloud

providers include traffic filtering, conventional IDS, browser integrity checking, and visitor whitelisting/blacklisting[103].

Each provider offers various methods of security therefore, consumers must install those not provided. Instead, then relying on the hosting provider, many firms conduct their own security[103].

This strategy creates hard-to-audit cloud security solutions[104]. The author's software defined perimeter (SDP) is a revolutionary cloud security technique that allows application owners to build perimeter features where needed. SDP conceals internal assets from external users, like fixed perimeter[104]. SDP offers finer grained control of the logical perimeter, rejecting untrusted devices that move into the physical organization (e.g., BYOD) and including trusted assets that stay outside (e.g., cloud-hosted services).

The researcher approach constantly modifies perimeter design to facilitate deception. Deception Honeynets Researchers utilize honeypots to learn about new attack strategies[90]. Honeypots are useful since they generate nothing. Honeypots provide no service or content thus real users have no need to utilize them. So, all traffic is harmful. Which therefore make the functionality separates server production and attack traffic[90].

It also allows for wider interaction with an opponent, as an unprotected asset may be misused[13]. If the attacker succeeds, more may be known about his intentions than if the firewall blocks his traffic. Honeypots and honeynets increase a network's attack surface to detect threats[13]. According to an author., this strategy may impose a lot of overhead and divert IT resources from output[13]. Active attacks must be controlled and supervised, and anyone lacking this expertise may incur liabilities via misconfigured phony assets. Enterprises won't use this technology until it's seamlessly integrated and maintained.

BotMiner combines connectivity and activity aspects to identify malware. Real and Vargas discovered hostile bots by passively monitoring subnet connectivity and activity[105]. BotMiner's inventors used X-means clustering on the C- and A-planes and cross-plane correlation to detect six of eight botnets on the test network[105]. The authors admit that their "A plane clustering approach has fairly weak cluster features" and that the observed A-plane activities were not particular to botnets, leading to "many false positives"[105].

A honeypot researcher carried out Empirical Assessment of Cyber Deception Masking, repackaging, dazzling, mimicking, inventing, and decoying all been utilized in kinetic military actions throughout history[106]. Another author said cyberspace taxonomy and deception for cyberdefense are growing[106]. Also, the author. has explored and developed this strategy that led to the discovery a topology-faking approach and another author created a deception paradigm to better analyze human, computer, network, and human-computer deceptions. Cyber security games have been used to test honeypot effectiveness, illustrating how different settings might compel attackers to adapt. The researcher's Honeypot research spawned deceptive cyberdefenses. Honey-patches supply an attacker with a false vulnerability and send them to a honeypot[106].

Researcher made real systems seem like honeypots that can gather intelligent information of the attackers[107]. This notion is significant because there is a limit to what can be done to make fraudulent systems seem more authentic (before they become real), and making actual systems look fake helps attackers understand which systems are valuable. Honeypots and deceptive studies have evolved.

The previously mentioned research examines decoy systems, few cyber deception studies have been done, and those that have lacked rigorous controls or large sample sizes. Larger studies use Internet strangers[107]. This study compared stage one malware to stage two human attacks based on typos. While they knew the dictionaries used in dictionary assaults in the wild, they discovered that honeypot attackers were script kiddies who didn't grasp UNIX access rights or erase history files. Deception on internet-facing network nodes creates adversarial behaviour, but lacks controlled study. Researcher assert no participant interviews or reports. Using technology-degreed students in controlled research is one option. Other researchers stated that the participant pool isn't indicative of proficient cyber attackers[107]. Students lack opponents' skills and reasoning. The employ the closest analogue to hostile cyber adversaries (red teamers) and more participants to boost statistical power and reliability[107]. Several projects are ongoing to experimentally study cyber deception.

These were suggested by a researcher and categorized into three rigorous solutions.

1) Simplifying cyber settings for non-experts.

2) Creating human-behavior-based simulators.

3) Using qualified volunteers.

Another researcher suggested that non-experts may understand realistic cyber-attack scenarios. Simulation programs like HackIT, and the network monitoring and attacks were replicated by another author. CS undergrads attacked honeypots more than real computers. Insufficient participants ruled out relevance. In FlipIt, researchers have tried to create models more lifelike[107]. In this two-player game, players flip resources to control them. Amazon Mechanical Turk recruited 155 participants to take the Short Dark Triad

personality test[107]. Another author identified attacker-defense strategy differences using competent volunteers in controlled experiments and assess the deception strategy's usefulness in providing cyber security to the Open Educational Resources backend[107]. Moonraker Study analyses host-based deception. The study's design contains a cover story, so participants are duped. While the Tularosa Study concentrated on decoys, other study revealed comparable results[107].

Ettinger uses honeypots to trap and observe intruders. This method is well-researched. Some technologies spoof network topology. Others have constructed a deception framework to analyze human, machine, or human-computer network deceptions. Cyber security games have been used to test honeypot effectiveness, illustrating how different settings might compel attackers to adapt[152]. Fake honeypots were constructed based on honeypots' efficacy[107].

The author explores the threat presented by attackers and how it might be minimized by developing a model[107]. The author didn't consider fake deceit from attackers[107].
This was said by another author to compared Network address shuffling to honeypots[117]. Embedded network address shuffling devices make homogenous networks seem varied and need less maintenance than honeypots. Shuffling network addresses hides the network's topology from an attacker. Little study has shown the efficacy of Network address shuffling systems, especially for recognizing attacker-generated bogus IP addresses[108].

Another research gap on Summers "Guideto.'s Implementer's Deception Technologies by SANS Institute of info Tech" on Network Address Shuffling is the attacker's ability to develop multiple IP addresses and transmit them to the existing network topology to

deceive the real deception technique from detecting the original IP address[108]. This might harm the network and confuse the IP's MAC address. Few trials have been done, and none had good experimental controls or a large enough sample size[108]. Larger research includes unknown internet users or cyber-related pupils. Cisco claimed neither participant pool is good for cyber attackers. Deception on internet-facing nodes produces aggressive behaviour, but it's unmanaged, confusing, and doesn't enable complaints or interviews[108]. Students lack opponents' skills and reasoning. The study utilizes red teams, the closest scientific counterpart of hostile cyber attackers, bring enough to guarantee statistical power as well as reliability[108].

In the research conducted by some honeypot authors on "Detecting's Attacks using Cyber Deception," honeypots are not effective enough for mainstream deployment, are difficult to construct and maintain, and are simple for attackers to discover[108]. Deception vs. honeypots: what's the difference? Deception involves more than simply honeypots, and technology has progressed; orchestration and virtualization are now commodities, enabling for previously unimaginable things[108]. Even five years ago, if one wanted to build up a network of honeypots, one had to manually disperse them between servers or find another means to make them appear credible. Scripts were needed to establish a honeypot on demand[108]. One couldn't automatically distribute Honeytokens throughout an organization's network. The next generation of cyber deception enables organizations to integrate deception aspects into their current security solutions to harness their potential. Because of deception components, the instruments become more useful[108].

Endnotes

[1]A. Abdulrahman, M. Ishaq, A. Fatima, A. Atika & Y. Suberu. "*A Proposed Improved Captcha Based Intrusion Detection Model*"**, Journal of Advanced Science and Optimization Research** Vol. 27, No.9, 2023 ISSN 2418-9325

[2]A. Ahmim, L. Maglaras, M. Ferrag, M. Derdour & H. Janicke. "*A Novel Hierarchical Intrusion Detection System Based on Decision Trees and Rules-based Models*". In 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), 2019, 228-233, DOI: 10.1109/DCOSS.2019.00059

[3]A. Arkhipova & D. Karevskiy. *"Honeypot as a Tool for Creating an Effective Secure System".* Novosibirsk State Technical University in Digital Technology Security Digital Technology Security 2021 ; https://doi.org/10.17212/2782-2230-2021-2-122-135

[4]A. Christin, C. Giselle, A. Wesam, A. Abu & S. Maha. "*A Comprehensive Survey for IoT Security Datasets Taxonomy, Classification and Machine Learning Mechanisms.* **Computers & Security,** 2023, 103283, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2023.103283.
(https://www.sciencedirect.com/science/article/pii/S0167404823001931)

[5]A. Elkosairy & A. Marianne. "*A New Web Deception System Framework"*, Conference: 2018 1st International Conference on Computer Applications & Information Security (ICCAIS) 2018, DOI: 10.1109/CAIS.8442027,

[6]A. Mishra & Sanjay K. Jain. "*A Survey on Question Answering Systems with Classification*". **Journal of King Saud University-Computer and Information Sciences** 28.3 2016, pp. 345–361. https://doi.org/10.1016/j.jksuci.2014.10.007

[7]A. Mudgal & S. Bhatia. **"***Spark-Based Network Security Honeypot System: Detailed Performance Analysis***"** Article, Dec 2022, **International Journal of Safety and Security Engineering.** 12. 2022, 737-743. 10.18280/ijsse.120610.

[8]A. Pashaei, Mohammad E. Akbari, Mina Z. Lighvan & C. Asghar. *"Early Intrusion Detection System using Honeypot for Industrial Control Networks"*, Results in Engineering, Volume 16, 2022, 100576, ISSN 2590-1230, https://doi.org/10.1016/j.rineng.2022.100576.

[9]A. Riancho. W3AF USER GUIDE. Available at: URL: http://cyber. lockheedmartin. com/hubfs/Gaining the Advantage Cyber Kill Chain. 26, 2021.

A. Schlenker, O. Thakoor, H. Xu, F. Fang, M. Tambe, L. Tran-Thanh, P. Vayanos & Y. Vorobeychik. "*Deceiving Cyber Adversaries: A Game Theoretic Approach* 17th International Conference on Autonomous Agents and Multiagent Systems, AAMAS Volume: 2, 2018 pp. 892–900.

[10]A. Shah. "*Evaluating Network Forensics Applying Advanced Tools*". **International Journal of Advanced Engineering, Management and Science**, Vol 9 No 4 2023, http://journal-repository.theshillonga.com/index.php/ijaems/article/view/6178

[11]A. Waqas, A. Muhammad, N. Sabreena & W. Farhana. "*Detection and Analysis of Active Attacks using Honeypot*". **International Journal of Computer Applications** (0975 – 8887) Volume 184 – No. 50, 2023 IJCATM: www.ijcaonline.org

[12]A. Yaser. "*Improving Intrusion Detection Systems Using Artificial Neural Networks*". **ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal**, Vol. 7 No. 1 2018 https://doi.org/10.14201/ADCAIJ2018714965

[13]B. Gupta & A. Gupta. "*Assessment of Honeypots: Issues, Challenges and Future Directions*". **International Journal of Cloud Applications and Computing (IJCAC)** 8(1) 2018 |Pp 34 DOI: 10.4018/IJCAC.2018010102

[14]B. Mphago, & S. Mpoeleng. "*Deception in Web Application Honeypots: Case of Glastopf*". **International Journal of Cyber-Security and Digital Forensics**, 6(4), 2017, pp. 179-185; DOI: 10.17781/P002304

[15]B. Paul & M. Rao. "*Zero-Trust Model for Smart Manufacturing Industry*". **Applied Sciences Journal.** 13(1) 2023 :221. https://doi.org/10.3390/app13010221

[16]B. Sara, C. Mauro, P. Luca & P. Pier. "*Social Honeypot for Humans: Luring People through Self-Managed Instagram Pages*". **Journal of Social and Information Networks, cs.SI), Artificial Intelligence (cs.AI), Cryptography and Security (cs.CR)** 2023 https://doi.org/10.48550/arXiv.2303.17946

[17]B. Temmie, V. Andrew, J. Kimberly, W. Ferguson, B. Sara, F. Daniel. & E. Kristin. "*The Moonraker Study: An Experimental Evaluation of Host-Based Deception*". In Hawaii International Conference on System Sciences (HICSS), Maui, Hawaii, 2020, DOI:10.24251/HICSS.2020.231.

[18]B. Abbaschian, S. Daniel & A. Elmaghraby. "*Deep Learning Techniques for Speech Emotion Recognition, from Databases to Models*", Computer Science and Engineering Department, University of Louisville, Louisville, KY 40292, USA, 2021, 21(4), 1249; https://doi.org/10.3390/s21041249

[19]C. Chou, C. Wu, K. Lu, L. Hsien & J. Li. *"Modbus Packet Analysis and Attack Mode for SCADA System*" **Journal of ICT, Design, Engineering and Technological Science**. 2018, 30-35. 10.33150/JITDETS-2.2.1.

[20]C. Kai, W. Zhan, L. Dongkun & R. Mu. *"The TaintDroid Based Honeypot Monitoring System for Embedded Device"*, **Journal of Physics Conference Series** 2203 (1):012077, 2022, DOI: 10.1088/1742-6596/2203/1/012077.

[21]C. Kai, W. Zhan, Z. Chengcheng & M. Haohua. *"The Research on Network Function Virtualization Based Network Honeypot"*, Proceedings of the 12th International Conference on Computer Engineering and Networks, 2022, DOI: 10.1007/978-981-19-6901-0_156,

[22]C. Kuan, L. I-Hsien & J. Li. *"Honeypot System of SCADA Security Survey",* Proceedings of International Conference on Artificial Life and Robotics 23:2018 pp 444-447, DOI: 10.5954/ICAROB.2018.OS8-8

[23]C. Sakama, M. Caminada, & A. Herzig. "*A Formal Account of Dishonesty*," **Logic Journal of IGPL,** vol. 23, 2015, no. 2, pp. 259–294, https://doi.org/10.1093/jigpal/jzu043

[24]D. Akshat, B. Anchit, A. Nihal & D. Sumithra. *"HONEYPOT: Intrusion Detection System"* **International Journal of Education Science Technology and Engineering** 3(1): 2020 pp 13-18, DOI: 10.36079/lamintang.ijeste-0301.66

[25]D. Danilov, T. Ovasapyan, D. Ivanov, A. Konoplev & D. Moskvin. *"Generation of Synthetic Data for Honeypot Systems Using Deep Learning Methods"*, Automatic Control and Computer Sciences, 2023, 56(8):916-926, DOI: 10.3103/S014641162208003X

[26]D. Rajesh, Thariq M. Hussan, B. Sri. Vastav. *"Network Protection Using Honeypots"*, **International Journal of Innovative Technology and Exploring Engineering** (IJITEE), Volume-9 Issue-6, 2020 ISSN: 2278-3075 (Online)

[27]D. Velasco & G. Rodriguez. *"A Review Of The Current State Of Honeynet Architectures And Tools"*, **International Journal of Security and Networks** 2017, pp 255-272, DOI: 10.1504/IJSN.10009165

[28]D. Zhang, F. Gang, S. Yang & S. Dipti. "*Physical Safety and Cyber Security Analysis of Multi-Agent Systems: A Survey of Recent Advances*." **IEEE/CAA Journal of Automatica** Sinica 8, no. 2, 2021: 319-333, DOI: 10.1109/JAS.2021.1003820

[29]D. Zielinski & Hisham A. Kholidy . *"An Analysis of Honeypots and their Impact as a Cyber Deception Tactic"*, 2022, DOI: 10.48550/arXiv.2301.00045.

[30]David P. Fidler. "*Just & Unjust War, Uses of Force & Coercion: An Ethical Inquiry with Cyber Illustrations*", Daedalus Vol. 145, No. 4, 2016, pp. 37-49 https://**www.jstor.org/stable**/24916782

[31]E. Abiodu, J. Aman & I Abiodu. "*A Comprehensive Review of Honey Encryption Scheme"*. **Indonesian Journal of Electrical Engineering and Computer Science**, Vol. 13, No. 2, 2019, pp. 649~656 ISSN: 2502-4752, DOI: 10.11591/ijeecs. v13.i2.

[32]E. Fujisaki. "*All-But-Many Encryption*", **Journal of Cryptology,** 31 2018**,** 31**,** pages 226–275, https://doi.org/10.1007/s00145-017-9256-x

[33]E. Iasiello. "*What is the Role of Cyber Operations in Information Warfare*"? **Journal of Strategic Security,** vol. 14, No 4, 2021, pp. 72-86, https://www.jstor.org/stable/48633489

[34]E. Morales, C. Rubio & A. Doupé. "*HoneyPLC: A Next-Generation Honeypot for Industrial Control System***s**", CCS '20: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security 2020, Pages 279–291https://doi.org/10.1145/3372297.3423356

[35]E. Zavadskii. & D. Ivanov. *"Counteracting Information Threats Using Honeypot Systems Based on a Graph of Potential Attacks"*, Automatic Control and Computer Sciences 56(8): 2023, 964-969, DOI: 10.3103/S0146411622080260

[36]E. Zavadskii.& D. Ivanov. *"Implementation of Honeypot Systems Based on the Potential Attack Graph"*, Automatic Control and Computer Sciences, 55(8):2021, 1194-1200, DOI: 10.3103/S0146411621080460,

[37]Eirini A. Anthi, W. Lowri, R. Matilda R, B. Pete, & W. Adam. "*Adversarial Attacks on Machine Learning Cybersecurity Defenses in Industrial Control Systems*." **Journal of Information Security and Applications** 58 2021: 102717, https://doi.org/10.1016/j.jisa.2020.102717

[38]F. Kimberly, F. Sunny, M. Justin, & M. Maxine. "*Game Theory for Adaptive Defensive Cyber-Deception*". Proceedings of the 6th Annual Symposium on Hot Topics in the Science of Security 2019 4 Pp 1–8 https://doi.org/10.1145/3314058.3314063

[39]F. McKee & D. Noever. *"Chatbots in a Honeypot World"*, Cryptography and Security (cs.CR); Computers and Society (cs.CY); Machine Learning (cs. LG), 2023, https://doi.org/10.48550/arXiv.2301.03771

[40]F. Wenjun, D. Zhihui, F. David & V. Victor. "*Enabling an Anatomic View to Investigate Honeypot Systems: A Survey*". **IEEE Systems Journal** Volume: 12, Issue: 4, 2017, DOI: 10.1109/JSYST.2017.2762161

[41]Fransiska S. Mukti & Muhammad R. Sukmawan. *"Integration of Low Interaction Honeypot and ELK Stack as Attack Detection Systems on Servers"* **Journal Penelitian Pos dan informatika** 11(1), 2021, DOI: 10.17933/jppi.v11i1.336, License CC BY-NC-SA 4.0

[42]G. Anand, T. Neha & H. Nayankumar. "*An Overview Of Honeypot Systems*, **International Journal Of Computer Sciences And Engineering,** 7(2): 2019, pp 394-397, Doi: 10.26438/Ijcse/V7i2.394397

[43]G. Elisavet, L. Athanasios, Panagiotis R. Grammatikis & Panagiotis G. Sarigiannidis. *"Protecting IEC 60870-5-104 ICS/SCADA Systems with Honeypots"*, IEEE International Conference on Cyber Security and Resilience (CSR), 2022, DOI: 10.1109/CSR54599.2022.9850329

[44]G. Tsochev, M. Sharabov & A. Georgiev. "*Using Machine Learning Reacted with Honeypot Systems for Securing Network*", International Conference Automatics and Informatics (ICAI), 2021, DOI: 10.1109/ICAI52893.2021.9639590

[45]H. Chie. "*Using The Modified Diffie-Hellman Problem to Enhance Client Computational Performance in A Three-Party Authenticated Key Agreement*". Arab. Journal. Science. Engineering. 43 (2), 2018, pp 637–644. https://doi.org/10.1007/s13369-017-2725-6

[46]H. Muhammad, Olumide B. Longe & Adebisi A. Baale. "*Towards the Development of a Machine Learning Enhanceed Framework for Honeypot and CAPTCHA Intrusion Detection Systems*" **Advances in Multidisciplinary and scientific Research Journal Publication** 2022, DOI: 10.22624/AIMS/ACCRABESPOKE2022/V34P4,

[47]H. Yuan, X. Changyou, D. Ke, Z. Guomin & S. Lihua. "*A Differential Privacy Based Multi-Stage Network Fingerprinting Deception Game Method*", **Journal of Information Security and Applications**, Volume 74, 2023, 103460, ISSN 2214-2126, https://doi.org/10.1016/j.jisa.2023.103460.(https://www.sciencedirect.com/science/article/pii/S2214212623000443)

[48]H. Zhou, C. Dong, R Wu, X. Xu & Z. Guo. "*Feature Fusion Based on Bayesian Decision Theory for Radar Deception Jamming Recognition"*, IEEE Access, vol. 9, 2021, pp. 16296-16304, doi: 10.1109/ACCESS.2021.3052506.

[49]I. Gbenga, & M. Yasser. *"Systematic Review of Graphical Visual Methods in Honeypot Attack Data Analysis* ", **Journal of Information Security**, 2022, 13, 210-243 https://www.scirp.org/journal/jis ISSN Online: 2153-1242 ISSN: 2153-1234

[50]J. Basak, M. Gutierrez, S. Curtis, C. Kamhoua, D. Jones, B. Bosansky & C. Kiekintveld. *"An initial study of targeted personality models in the flipit game*". International Conference on Decision and Game Theory for Security, vol 11199, 2018 pp 623–636 https://doi.org/10.1007/978-3-030-01554-1_36

[51]J. Dansana, K. Kabat. & K. Pattnaik. *"A Novel Optimized Perturbation-Based Machine Learning for Preserving Privacy in Medical Data.* **Wireless Personal Communication** 2023. https://doi.org/10.1007/s11277-023-10363-x

[52]J. Kimberly & W. Ferguson. "*The Tularosa Study: An Experimental Design and Implementation to Quantify the Effectiveness of Cyber Deception*", Proceedings of the 52nd Hawaii International Conference on System Sciences 2019, http://hdl.handle.net/10125/60164

[53]J. Logeshwaran, G. Ramesh & V. Aravindarajan. "*A Secured Database Monitoring Method to Improve Data Backup and Recovery Operations in Cloud Computing*". **BOHR International Journal of Computer Science** Vol. 2 No. 1 2023: https://journals.bohrpub.com/index.php

[54]Jarot S Suroso & Caesario P. Prastya. "*Cyber Security System with SIEM And Honeypot In Higher Education*", IOP Conference Series Materials Science and Engineering 874(1): 2020, 012008, DOI: 10.1088/1757-899X/874/1/012008

[55]Jorge B. Garcia. *"Creation of a High-Interaction Honeypot System Based-on Docker Containers"*, Conference: 2021 Fifth World Conference on Smart Trends in Systems Security and Sustainability (WorldS4), 2021, DOI: 10.1109/WorldS451998.2021.9514022

[56]K. George, N. Grigoris, K. Irodotos & L. Sotiris I. *"HoneyChart: Automated Honeypot Management over Kubernetes"*, ESORICS 2022 International Workshops, vol 13785. 2022, https://doi.org/10.1007/978-3-031-25460-4_18.

[57]K. Hoffpauir, N. Markle, C. Meadows & J. Pittman. "*A Taxonomy for Dynamic Honeypot Measures of Effectiveness*". Cryptography and Security (cs.CR) 2020, https://doi.org/10.48550/arXiv.2005.12969

[58]K. Lu, I. Liu & Jung-S. Li. *"Honeypot System of SCADA Security Survey*", Proceedings of International Conference on Artificial Life and Robotics, , 23:2018, Pp 444-447, DOI: 10.5954/ICAROB.2018.OS8-8

[59]K. Lu, I. Liu, J. Liao & C. Li. *"Evaluation and Build to Honeypot System about SCADA Security for Large-Scale IoT Devices"*, **Journal of Robotics Networking and Artificial Life** 6(3), 2019, DOI: 10.2991/jrnal.k.191202.008, License CC BY-NC

[60]K. Meatasit, E. Hiroshi & O. Hideya. **"***SDNHive: A Proof-of-Concept SDN and Honeypot System for Defending Against Internal Threats***",** Conference: ICCNS 2021: 11th International Conference on Communication and Network Security, 2021, DOI: 10.1145/3507509.3507511

[61]K. Patel & D Chudasama. "*National Security Threats in Cyberspace*", **National Journal of Cyber Security Law**, Volume 4, Issue 1, 2021 pp 109–114 DOI: 10.37591/NJCSL, http://lawjournals.celnet.in/index.php/njcsl/index 2(2),

[62]K. Pooja & K. Ankit. "*A comprehensive study of DDoS attacks over IoT network and their countermeasures*", **Computers & Security,** Volume 127, 2023, 103096, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2023.103096. **(https://www.sciencedirect.com/science/article/pii/S0167404823000068)**

[63]K. Sakthidasan & B. Kim. *"Deep Learning Based Energy Efficient Optimal RMC-CNN Model for Secured Data Transmission and Anomaly Detection in Industrial IOT"***, Sustainable Energy Technologies and Assessments,** Volume 56, 2023, 102983, ISSN 2213-1388, https://doi.org/10.1016/j.seta.2022.102983. **(https://www.sciencedirect.com/science/article/pii/S2213138822010311)**

[64]K. Santhosh, M. Selvi & A. Kannan. "A Comprehensive Survey on Machine Learning-Based Intrusion Detection Systems for Secure Communication in Internet of Things" **Journal of Computational Intelligence and Neuroscience,** 2023 Article ID 8981988 | https://doi.org/10.1155/2023/8981988

[65]L. Brooke & M. Weizhi. *"A survey of deep learning-based intrusion detection in automotive applications"* **Expert Systems with Applications**". Vol. 221, 2023, 119771, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2023.119771. **(https://www.sciencedirect.com/science/article/pii/S0957417423002725)**

[66]L. Gururaj, H. Swathi, R. & Trupti. "*Analysis of Preventive Measures Against DDoS Attacks in Smart Grid*. **Journal of the Institution of Engineers** 104, 2023 pp297–303 https://doi.org/10.1007/s40031-022-00844-1

[67]L. Huang & Q. Zhu. "*Adaptive Honeypot Engagement Through Reinforcement Learning of Semi-Markov Decision Processes*", International Conference on Decision and Game Theory for Security 2019 pp. 196-216. DOI https://doi.org/10.1007/978-3-030-32430-8_13

[68]L. Irini, S. Shreyas, V. Emmanouil & G. Dimitris. "A Decentralized Honeypot For Iot Protocols Based On Android Devices", **International Journal of Information Security** 21(1), 2022 DOI: 10.1007/s10207-022-00605-7

[69]L. Irini, S. Shreyas, V. Emmanouil & G. Dimitris. "*A Decentralized Honeypot for IoT Protocols Based on Android Devices"*, **International Journal of Information Security** 21(1), 2022, DOI: 10.1007/s10207-022-00605-7

[70]L. Kyungroul, L. Jaehyuk & Y. Kangbin. "*Classification and Analysis of Malicious Code Detection Techniques Based on the APT Attack*". **Journal of Applied Sciences,** Volume 13 Issue 2023 5 10.3390/app13052894

[71]L. Matthew, C. Christopher, & F. Hiroshi. "*A Survey: Recent Advances and Future Trends in Honeypot Research*". **International Journal of Computer Network and Information Security,** 4. 2018 doi: 10.5815/ijcnis.2012.10.07

[72]L. Ning, C. Bo & L. Ziyan. *"IoT Honeypot Scanning and Detection System Based on Authorization Mechanism"* International Conference of Pioneering Computer Scientists, Engineers and Educators 2021 pp 217–228 DOI: 10.1007/978-981-16-5943-0_18, In book: Data Science,

[73]L. Roye. "Incorporating a Honeyfarm with Mlffnn IDS for Improving Intrusion Detection". **International Journal of Advanced Research in Computer Science**. Vol. 14 Issue 1, 2023 pp 1-4.

[74]L. Suhyeon, C. Kwangsoo & K. Seungjoo. *"Do You Really Need to Disguise Normal Servers as Honeypots?"* EEE Military Communications Conference (MILCOM) on Cryptography and Security (cs.CR); Computer Science and Game Theory (cs.GT) 2022, DOI: 10.48550/arXiv.2210.17399

[75]M. Ala, O. Ibrahim, A. Ashraf, A. Shadi, Q. Fatima, A. Dena, A. Aseel & A. Laith. "*Simulation and Analysis Performance of Ad-Hoc Routing Protocols Under DDOS Attack and Proposed Solution*". **International Journal of Data and Network Science** Volume 7 Issue 2 2023 pp. 757-764, 3ISSN 2561-8156 (Online) - ISSN 2561-8148 (Print) DOI: 10.5267/j.ijdns.2023.2.002

[76]M. Anupama & A. Ammar. *"Malware Detection Techniques: A Comprehensive Study",* **An International Interdisciplinary Journal**, Vol. 01, No. 01, 2023 pp 1 – 5

[77]M. Ateek. M. Jafirkhan, M. Shubhangi. *"Data Security using Honeypot System"*, **International Research Journal of Engineering and Technology (IRJET)** e-ISSN: 2395-0056Volume: 05 Issue: 03 2018 www.irjet.net p-ISSN: 2395-007

[78]M. Bringer, C. Chelmecki & H. Fujinoki. "*A Survey: Recent Advances and Future Trends in Honeypot Research*," **International Journal of Computer Network and Information Security**, vol. 4, no. 10 2012 pp. 63, DOI: 10.5815/ijcnis.2012.10.07MECS (http://www.mecs-press.org/)

[79]M. Chandane & S. Vaishali. **"***Efficacy Measuring Framework for the Assessment of Dynamic Honeypot*", International Conference on Advances in Computing, Communication, and Control (ICAC3), 2021 DOI**:** 10.1109/ICAC353642.2021.9697296,

[80]M. Kang & L. Kang. "*Intrusion Detection System Using Deep Neural Network for In-Vehicle Network Security*". **PLoS ONE Journa**l 11(6) 2016 e0155781. https://doi.org/10.1371/journal.pone.0155781

[81]M. Sam, G. Vasileios, G. Benjamin & P. Nicholas. Race, *"Don't get Stung, Cover your ICS in Honey: How do Honeypots fit within Industrial Control System Security",* Computers & Security 114(4):102598, 2022, DOI: 10.1016/j.cose.2021.102598

[82]M. Stefan. *"Honeypot Implementation in a Cloud Environment*", **Journal of Cryptography and Security**, 2023, https://doi.org/10.48550/arXiv.2301.0071

[83]N. Abdul, Z. Muhammad & S. Suherman. *"Analysis and Implementation of Honeyd as a Low-Interaction Honeypot in Enhancing Security Systems"*, **Randwick International and Social Dcience Journal** Vol. 2 No. 1 2021, DOI: 10.47175/rissj. v2i1.209,

[84]N. Agrawal & T. Shashikala. "*The Performance Analysis of Honeypot Based Intrusion Detection System for Wireless Network",* , **International Journal of Wireless Information Networks** 24(1) 2017, DOI: 10.1007/s10776-016-0330-3

[85]N. Kamel, E. Mohamed, L. Youssef & T. Raja. *"A Smart Agent Design for Cyber Security Based on Honeypot and Machine Learning*", Hindawi Security and Communication Networks, ID 8865474, 2020, https://doi.org/10.1155/2020/8865474

[86]N. Thangarasu & A. Selvakumar. "*Improved Elliptical Curve Cryptography and Abelian Group Theory to Resolve Linear System Problems in Sensor-Cloud Cluster Computing"*. 22. 2019, 13185–13194 (2019). https://doi.org/10.1007/s10586-017-1573-1

[87]N. Vincent, K. Mohamed, A. Eric & H. Matthieu. "*Setup and Deployment of a High-Interaction Honeypot: Experiment and Lessons Learned*" **Journal in Computer Virology,** 7(2):2011, 143–157. doi: 10.1007/ s11416-010-0144-20

[88]O. Abdulganiyu, T. Ait. & Y. Saheed. *"A Systematic Literature Review for Network Intrusion Detection System (IDS)".* **International Journal of Information Security** *2023* https://doi.org/10.1007/s10207-023-00682-2

[89]O. Duan, E. Al-Shaer, M. Islam, & H. Jafarian. "*Conceal: A Strategy Composition for Resilient Cyber Deception-Framework, Metrics and Deployment*," in IEEE Conference on Communications and Network Security (CNS), 2018, pp. 1–9, DOI: 10.1109/CNS.2018.8433196.

[90]P. Abbasgholi, Mohammad E. Akbari, Mina Z. Lighvan & C. Asghar. "*Machine Learning-Based Early Intrusion Detection System in Industrial LAN Networks Using Honeypots"* 2021, DOI: 10.21203/rs.3.rs-1122586/v1, License, CC BY 4.0.

[91]P. Dimitrios, Panagiotis G. Sarigiannidis, L. Athanasios & S. Ilias. *"A Novel and Interactive Industrial Control System Honeypot for Critical Smart Grid Infrastructure"*, IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 2019, DOI: 10.1109/CAMAD.2019.8858431

[92]P. Sharma, S. Kapoor & R. Sharma. "*Ransomware Detection, Prevention and Protection in IoT Devices Using ML Techniques Based on Dynamic Analysis Approach*". **International Journal of System Assurance Engineering and Management** 14, 2023 287–296. https://doi.org/10.1007/s13198-022-01793-0

[93]R Susanti, A. Muhammad & W Prabowo "*Implementation Intrusion Prevention System* (IPS) OSSEC dan Honeypot Cowrie" **Jurnal Sisfokom (Sistem Informasi dan Komputer)** 11(1):2022, 73-78, DOI: 10.32736/sisfokom.v11i1.1246, License CC BY 4.0

[94]R, Dušan & K Tomaž. *"Artificial Intelligence for Cybersecurity: Literature Review and Future Research Directions",* Information Fusion. Vol. 97, 2023. https://doi.org/10.1016/j.inffus.2023.101804.

[95]R. Amal & P. Venkadesh. **"**Review of Cyber Attack Detection: Honeypot System"**.** Webology, Vol. 19, Number 1, 2022, DOI: 10.14704/WEB/V19I1/WEB19370

[96]R. Amal & P Venkadesh. "H-DOCTOR: *Honeypot Based Firewall Tuning for Attack Prevention***",** Measurement: Sensors, Vol, 25, 2023, https://doi.org/10.1016/j.measen.2022.100664**.(https://www.sciencedirect.com/science/article/pii/S2665917422002987)**

[97]R. Barbulescu & S. Duquesne. "*Updating Key Size Estimations for Pairings*". **Journal of Cryptology**. 32, 2019, pp 1298-1336. https://doi.org/10.1007/s00145-018-9280-5

[98]R. Shah & K. Rajapraveen. *"Secured Honeypots To Understand Attacks To Control Systems",* **International Journal for Science and Advance Research In Technology**, Volume 5 Issue 8 2019, ISSN [ONLINE]: 2395-1052

[99]R. Sulaiman & R. Masud. "*A Detailed Study on Web-Based-Honeypot to Propose Mitigation Framework in Web Application***". SSRN Electronic Journal,** 2019, DOI: 10.2139/ssrn.3723098

[100]R. Vishwakarma. "*A Honeypot with Machine Learning Based Detection Framework for Defending IoT Based Botnet DDoS Attacks*," 3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019, pp. 1019-1024, doi: 10.1109/ICOEI.2019.8862720.

[101]S Rejwan & R. Masud. A Detailed Study on Web-Based-Honeypot to Propose Mitigation Framework in Web Application. **SSRN Electronic Journal.** 2019**. DOI:**10.2139/ssrn.3723098**.**

[102]S. ABE, Y. Yohei, U. Yukako & H. Shinichi**.** *"Developing Deception Network System with Traceback Honeypot in ICS Network"***, SICE Journal of Control, Measurement, and System Integration** 11(4):2018 372-379, DOI: 10.9746/jcmsi.11.372

[103]S. Cooney, K. Wang, E. Bondi, T. Nguyen, P. Vayanos, H. Winetrobe, E. Cranford, C. Gonzalez, C. Lebiere, & M. Tambe. "*Learning to Signal in the Goldilocks Zone: Improving Adversary Compliance in Security Games*", Joint European Conference on Machine Learning and Knowledge Discovery in Database, 2019, pp 725–740, DOI: https://doi.org/10.1007/978-3-030-46150-8_42

[104]S. Ding & C. Louen "*Detection of Deception Attacks in Cyber Physical Systems Using Signal Shaping Part I - Basic Scheme", TechRxiv. Preprint 2023, DOI: 10.36227/techrxiv.22268434*

[105]S. Dubravko S & Tonimir K. *"Efficiency And Security of Docker Based Honeypot Systems"***,** International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2018, DOI: 10.23919/MIPRO.2018.8400212

[106]S. Ismael, Z. Tariq, & Samah J. Sabaa. "*Online Intrusion Detection System Using C4.5 Algorithm with Honeypot"***,** Journal of Engineering and Applied Sciences 15(5): 2019 1127-1132 DOI: 10.36478/jeasci.2020.1127.1132

[107]S. Kim's Lab, S. Lee & S. Kim. *"Do You Really Need to Disguise Normal Servers as Honeypots?*", Military Communications Conference (MILCOM) 2022, DOI: 10.1109/MILCOM55135.2022.10017586

[108]S. Morteza, N. Christelle, F. Kurt & B. Elias. "*A Comprehensive Survey of Recent Internet Measurement Techniques for Cyber Security*", **Computers & Security,** Vol. 128, 2023, 103123, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2023.103123. **(https://www.sciencedirect.com/science/article/pii/S0167404823000330)**

[109]S. Park, G. Li & J. Hong, "*A Study on Smart Factory-based Ambient Intelligence Context Aware Intrusion Detection System Using Machine Learning*," **Journal of Ambient Intelligence and Humanized Computing**, 11, 2018 1405–1412. https://doi.org/10.1007/s12652-018-0998-6

[110]S. Rajasoundaran, R. Maheswar & M. Akila. "Interleaved Honeypot-Framing Model with Secure MAC Policies for Wireless Sensor Networks", Smart Communication Protocols and Algorithms for Sensor Networks 22(20):8046, 2022, DOI: 10.3390/s22208046, License CC BY 4.0.

[111]S. Shreyas, Jens M. Pedersen & V. Emmanouil. "*Gotta Catches Em All: A Multistage Framework for Honeypot Fingerprinting*", Digital Threats: Research and Practice. 10.1145/3584976. 2023, DOI: 10.1145/3584976.

[112]S. Touch & J. Colin, "*A Comparison of an Adaptive Self-Guarded Honeypot with Conventional Honeypots",* Applied Sciences 12 (10): 5224, 2022, DOI: 10.3390/app12105224

[113]S. Vivek. "*Design & Implementation of Honeyd to Simulate Virtual Honeypots",* **IOSR Journal of Computer Engineering** 3(1): 2012, 28-34**,** 2012 DOI: 10.9790/0661-0312834

[114]T. Nisha & P. Dhanya. *"Insider Intrusion Detection Techniques: A State-of-the-Art Review".* **Journal of Computer Information Systems,** 2023, https://doi.org/10.1080/08874417.2023.2175337

[115]V. Danny & R. Glen. "*Ontology for Data Integration in Honeynet*". **European Journal of Military Studies** a Multidisciplinary Journal Vol. 13 No. 2 2023

[116]V. Sviatoslav, S. Vitalii, O. Ivan, K. Yevhenii & T. Ivan. "*A Model of Decoy System Based on Dynamic Attributes for Cybercrime Investigation*". **Eastern-European Journal of Enterprise Technologies,** 1(9 (121), 2023 6–20. doi.10.15587/1729-4061.2023.273363

[117]V. Veronica, R. Maria & G. Sebastian. *"Attacker Profiling Through Analysis of Attack Patterns in Geographically Distributed Honeypots".* **Journal of Cryptography and Security (cs.CR); Networking and Internet Architecture (cs.NI),** 2023 https://doi.org/10.48550/arXiv.2305.01346

[118]W. Ahmad & Alrashdan T. "The Effect of Using Honeypot Network on System Security "**International Journal of Data and Network Science** 6:2022 pp 1 – 6, DOI: 10.5267/j.ijdns.2022.5.010

[119]W. Ahmad, Muhammad A., Sabreena N & Farhana W. "*Detection and Analysis of Active Attacks using Honeypot*". **International Journal of Computer Applications** (0975 – 8887) Volume 184 – No. 50, 2023 IJCATM: www.ijcaonline.org

[120]W. Fan, Zhihui D, David F & Victor V. "*Enabling an Anatomic View to Investigate Honeypot Systems: A Survey*" **EEE Systems Journal,** vol. 12, no. 4, pp. 2018 3906-3919, doi: 10.1109/JSYST.2017.2762161.

[121]X. Yang, Jie Y, Hao Y, Ya K, Hao Z & Jinyu Z. "*A Highly Interactive Honeypot-Based Approach to Network Threat Management*", **Journal of Future Internet,** 2023, 15(4), 127; https://doi.org/10.3390/fi15040127

[122]X. Qiin, Frank J, Mingcan C & Robin D *"Hybrid cyber defense strategies using Honey-X: A survey*". Computer Networks, Volume 230, 2023, 109776, ISSN 1389-1286, https://doi.org/10.1016/j.comnet.2023.109776
(https://www.sciencedirect.com/science/article/pii/S1389128623002219)

[123]Y. Jasim. "*Improving Intrusion Detection Systems Using Artificial Neural Networks*". **ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal**, Vol. 7 No. 1 2018 https://doi.org/10.14201/ADCAIJ2018714965

Chapter Three

Methodology

3.1 Research Approach

This chapter discusses the details methods adopted in designing an Optimized web based Low-Level Interaction Glastopf Honeypot system for accurate detection of fake Honeypot using OMNET++ simulation. The research approach used in this research work is quantitative one which was chosen because of various number of attackers to be lunched on a vulnerable web application and how efficiently an enhanced Glastopf honeypot system could gather attackers' intelligent information which can be used for forensic analysis and investigation. The methodology adopted was firstly to study various deception technology tools and to develop a virtual simulation system where the configuration and implementation of Glastopf honeypot system was carried out for the purpose of testing its efficiency and detection of fake honeypot by the attackers.

The study includes detailed implementation of Optimized Glastopf Honeypots**.** During the research, a centralized logging and monitoring web-based server on Elasticsearch, Logstash & Kibana (ELK) stack was utilized. This logging system is designed to monitor live traffic and direct it to the honeypot to avoid traffic going towards the production system. Elasticsearch was used because it is able to achieve fast search responses instead of searching the text directly, it searches an index instead. Elasticsearch is designed to be scalable and distributed. Kibana is an open source (Apache Licensed) browser-based analytics and search dashboard for Elasticsearch that visualized the data to provide a better interpretation. It is used to visualize captured logs from compromised honeypots.

3.1.1 Glasstopf Honeypot

Glastopf is a popular type of honeypot system that is specifically designed to emulate web application vulnerabilities and attract attackers. This was chosen due to its flexibility and ability to be easily deployed in different environments. It allows security professionals to gain a deeper understanding of web application attacks and improve defenses by proactively identifying vulnerabilities and enhancing security measures.

However, it's important to note that managing a honeypot system like Glastopf requires expertise and careful monitoring, as there is a risk of actual attacks reaching the real network if proper isolation measures are not in place. Popular attack type emulation is in place like remote file inclusion, local file inclusion using virtual file system and POST requests to do injections in HTML. Glastopf basically emulates web services of any web production server and log all the attacks and exploits that end up in web services[1].

The technique utilized for this study is quantitative method with the use of OMNET++ simulation tool for the simulation of honeypot system. A honeypot system was built and setup with a baseline Linus OS to assess its efficiency and efficacy[2]. The Linux RedHat 7.3 baseline system provides a typical business server for SMTP, POP3, SSH, Telnet, and FTP installation and configurations. On Linux RedHat 7.3, a honeypot system simulator hosts web applications as a trap for attackers[7]. A Glastopf honeypots system was configured to send an automatic bell ring or SMS to the system administrator in event of an attack. After configuring honeypots, there was a demonstration of the experiment with web applications and exposing it to attackers to lunch a phishing attack to the honeypots simulation system, which allowed us to test the efficiency of the honeypot system[2].

The Glastopf honeypots is designed to simulates vulnerable web applications and captures any malicious activity directed towards it alert the system administrator. When an attacker tries to exploit the simulated vulnerabilities in Glastopf, their actions are recorded and analyzed for further investigation. This can provide valuable insights into current attack techniques, trends, and even the identification of new vulnerabilities[3]. This will also integrate with various log analysis tools and intrusion detection systems to enhance its effectiveness and detection of fake honeypot

3.1.2 Flowchart Sequence

3.1.2.1 ELK Stack

The ELK stack contains Elasticsearch, Logstash, and Kibana. Elasticsearch is a powerful open-source solution for any knowledge extraction problem. A developer can single handedly use this ELK stack and can solve any unstructured database problem easily. Logstash is a tool that take data as input, process it and output it in structured format to Elasticsearch. Logstash can manage any type log such that: system logs, webserver logs, error logs, and app logs. Kibana is dashboard of log-data. It provides flexibility to easily create pie charts, bar graphs, trend-lines, maps and scatter plots[4].

This project intends to develop and set up SSH Cloud honeypot and display the findings using the ELK Stack. Cloud honeypot and Data Visualization to avoid SSH assaults[4]. The designed architecture improves detection, protection, and attack reporting[4]. This design focuses on the three components listed in the section. (a) Detectors: This finds SSH assaults are simple to perform and hard to detect, hence SSH detection is key to mitigating them. Malicious system activity requires detection. Figure 3.1 shows the inner detection design, which focuses on brute force, dictionary, SSH port scan, and SSH penetration

assaults. It detects SSH brute-force, dictionary, port scan, and penetration assault[4]. After user or attacker enters login and password, the suggested design checks them. Before allowing access, the system verifies the user's legitimacy. If an attacker tries to log in more than five times, the system sends an email alert with some details. SSH employs public-key cryptography for user authentication and to encrypt host-to-host communication. If not the genuine user, the system will identify the nature of SSH assaults, whether brute-force, dictionary, port scan, or penetration attacks[4].

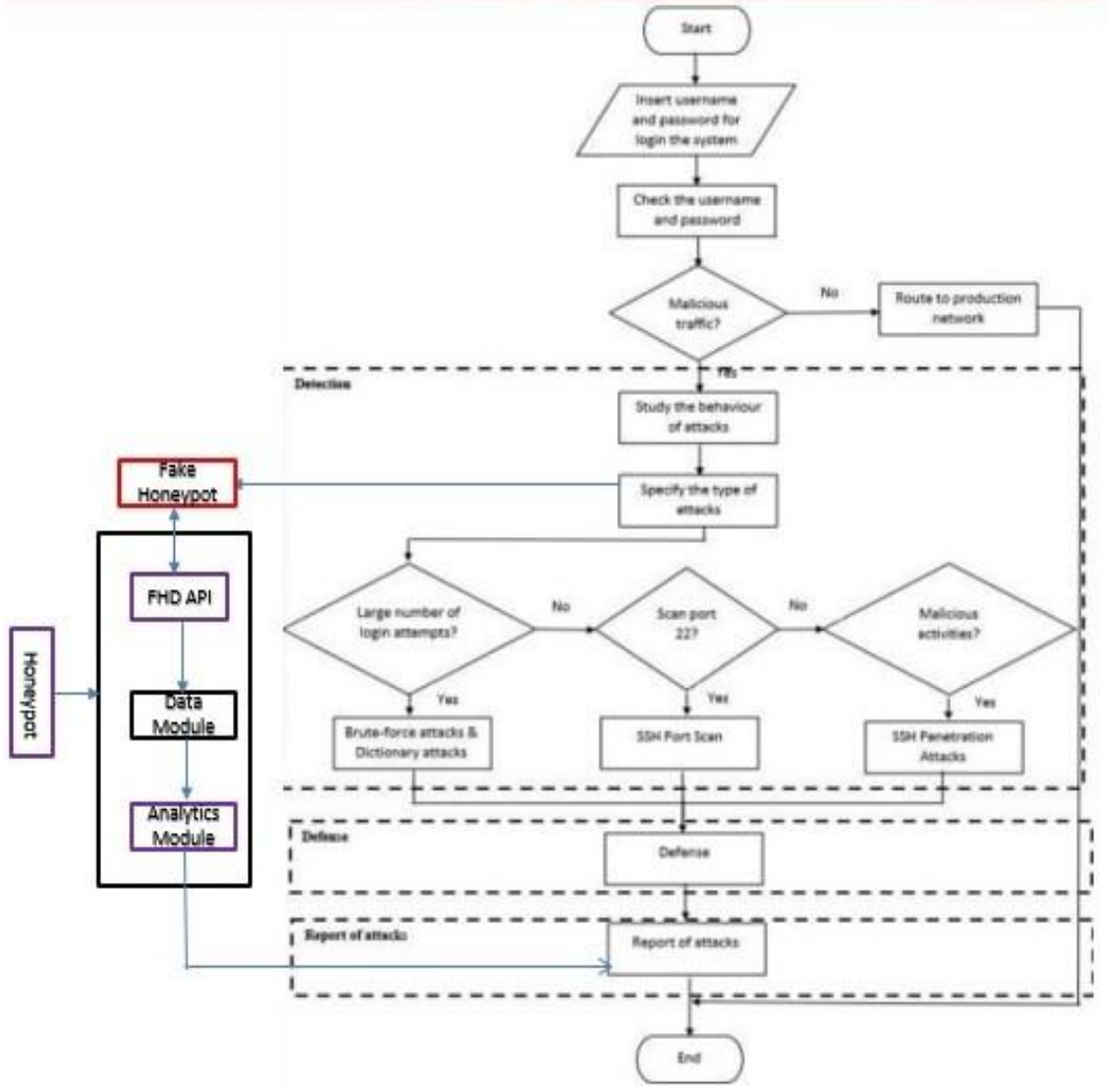


Figure 3.1 Sequential Flowchart Design[4]

3.2 Requirement Specifications

Honey assets must be accessible, realistic, measurable, ethical, and robust (ARMER). The followings explain the requirement that should be met:.

1. Accessible

When generating honey assets, make sure they can be accessed by the desired audience and researcher(s). Honey assets are best designed and deployed on systems hackers already use or can access easily[5]. Same for honey asset builders (the chosen platform must be accessible to the researcher that intends to carry out studies using honeypots). Webmail, social, and dating services are available[5].

2. Realistic

To be persuasive, honey assets must resemble real-world instances. Honey webmail accounts must seem legitimate while being fraudulent[19]. Content and presentation of honey assets should be carefully considered. Honey assets must leverage data sources that mimic real-user content[6]. Cybercriminals and other visitors will believe honey assets created from such material at first sight. This will eliminate possible bias if honey assets look "strange," which may alter visitors' behaviour. HoneyGen may be used to produce realistic decoy data[6].

3. Measurable

It's crucial to design honey assets so researchers can simply gather data and measure them[7]. This is done by combining intrinsic tools, such as Google Apps Script3 in Gmail accounts, with extrinsic tools, such as scripts built by the researcher to connect to honey assets and record activity[7]. It's worthless to construct and deploy honey assets if activity data can't be

collected from them. Honeypots are meant to fool attackers into thinking they are engaging with actual assets hence honeypot instrumentation must be "hidden"[7].

4. Ethical

Honey assets must be created and utilized ethically[8]. By separating potentially dangerous honeypot situations, the major ethical purpose for honey assets is to minimize damage to the target audience. The researcher shall maintain all activity data secure and not de-anonymize honey visits. If tests entail running real malware samples, care must be taken to ensure they don't affect internal or external parties[2]. Social honeypot researchers should protect the honey researcher's valuables. Honey assets and honeypot systems must limit damage to researchers, such as if their names are revealed during trials. Using VPNs and proxies in honeypot architecture is critical[8].

5. Robust

This thesis explores honey assets and honeypot systems, which need external platforms[9]. Studies on hacked webmail accounts use accounts hosted by a webmail provider, not under the researcher's direct control[9]. It's important to construct fault-tolerant honeypot systems and honey assets so external changes (webmail services, for example) don't influence trials. Sometimes it's difficult to construct a honeypot that adjusts to all external changes. The honeypot researcher must be ready to make tiny alterations to respond to external changes[9]. Adding modest modifications to honey programs that monitor a web page is one example. If the website's user interface changes, the honeypot researcher must update their script. In summary, a strong honeypot must be built and monitored closely throughout operation, with modest adjustments made as needed[9].

3.2.1 Development Tools

The following tools have been used in the development life cycle of the system:

1 **Glastopf**: It is the most powerful honeypot to analysis web application. It helps to emulate different vulnerability type like RFI, LFI, SQL injection, html injection etc.

2 **Sublime Text 2**: One of the powerful text editors. It is popular because of its user friendliness and unique feature that come with it.

3 W3AF: it is security scanner with addition to audit framework.

4 Archini: It is an open-source tool that helps the security professional to evaluate web application security[10].

5 **Oracle VM VirtualBox:** one of the popular virtual boxes that helps to create virtual environment in any operating system[10].

6 **Kali Linux OS:** It is the most popular and powerful operating system that helps the security professionals to perform pen testing and security audit with different tools and techniques.

7 **CodeLobster** Code Lobster is a great PHP development tool. It has smart auto-completion, an HTML examiner, DOM elements, code featuring, etc. WordPress, Drupal, Joomla, and Magento use CodeLobster. It simplifies PHP programming overall[11].

8 **Cloud9** Cloud9 is a cloud-based IDE that lets you write, execute, and inspect code using a program. It underlies C, C++, PHP, Ruby, Perl, Python, JavaScript, and Node.js. Cloud9 supports Terminal for order execution and cloning the development environment. In addition to smart auto-completion and an innovative

debugger, it lets developers pick from a variety of Runners, such as Ruby, Python, and PHP/Apache[12].

9 **Atom** is a revolutionary word processor. It supports Windows, MacOS, and Linux. The Atom is great since it can be edited without consulting the configuration manual[13]. Auto-complete helps one write code faster and smarter. Atom has a simple UI and distinct windows for each job or activity[13].

3.2.2 Hardware and Software Environment

The hardware methodology of the project will have minimal contribution since the honeypot will not possess any dependency on the hardware. However, since the honeypots will be used in the computer for the system development, the hardware part has been included, and the computer's hardware specifications are mentioned below. The honeynet system uses Linux virtual machines running on VMware Workstation version 12 with 12 GB RAM and 500 GB storage. The virtual machine-based honeynet features a centralized logging and automation server. A router, firewall, IDS, and Linux server host make up the honeypot system[14]. Attackers targeted the Linux server's Secure Shell (SSH) service.

3.2.3.1 Hardware Specifications

i. Disk Space	500GB
ii. Processor	Intel Core 2 Duo 4.5GHz 64 bit
iii. Memory	(RAM)8GB
iv. Display	XGA (1024 * 768 pixels)
v. Optical Drive	Internal / External DVD *8 drive
vi. Input Device	Keyboard and Mouse

3.2.3.2 Software Specifications

i. Operating System (OS) Linux RedHat 7.3

ii. Application software Honeypots simulator

iii. XAMP Server to host the PHP, Apache, and MySQL for the database for the purpose of social media account creation

iv. Python Object Oriented Programming Language

v. Office Application

vi. .NET Framework

3.2.3.3 Human ware Requirement

i. System Security Administrator

ii. System Users

iii. Hackers

iv. Webmail Account Users

v. Social Media Account Users

vi. Cloud Document Account Users

3.2.3.4. Technology Requirement

i. Honeypot System

ii. Nomadic Honeypot

iii. Hostage Honeypot

iv. OMNet++ Discrete event Simulator for honeypot Simulation

3.3 System Design

The illustration of the configuration, design, and deployment of the honeypot was done in the Linux operating system that is needed to be hosted in the virtual environment, e.g., VMware or Virtual box. The honeypot can be attacked by different types of exploitations with numerous scanning tools[15]. Honeypot has the potential to generate valuable information that can never be generated by any type of prevention system or intrusion detection system available in the market. The network administrators are capable of registering the alerts based on the information they get, and the administrators can become cautious against the possible attacks by the attackers. Hence, the administrators can have significant time so that the defense mechanism of the system can have strength[16].

Figure 3.2 depicts the system design architecture for the proposed honeypots system. The whole network is first secured by a firewall, then by a router, and compartmented data layers are separated from the network inside the company and from the network outside the firm's customers' or operations' network[17]. The organization's network is then safeguarded by the honeynet mechanism, which is a network of computers that participate in the honeypot architecture. IDS is used in the system for added security and detection. The monitoring control system manages the logs generated by the honeynet and also monitors any incoming network entries[17].

A Glastopf honeypots have been proposed and implemented to capture network attacks on three different services in this research. The honeypots implemented are presented as an enhancement to currently available honeypots in the field of system security. The presented honeypots have been integrated with the Log management model developed on the top of

ELK framework. This research will also discuss the issues associated with each honeypot and the security threat model of each honeypot.

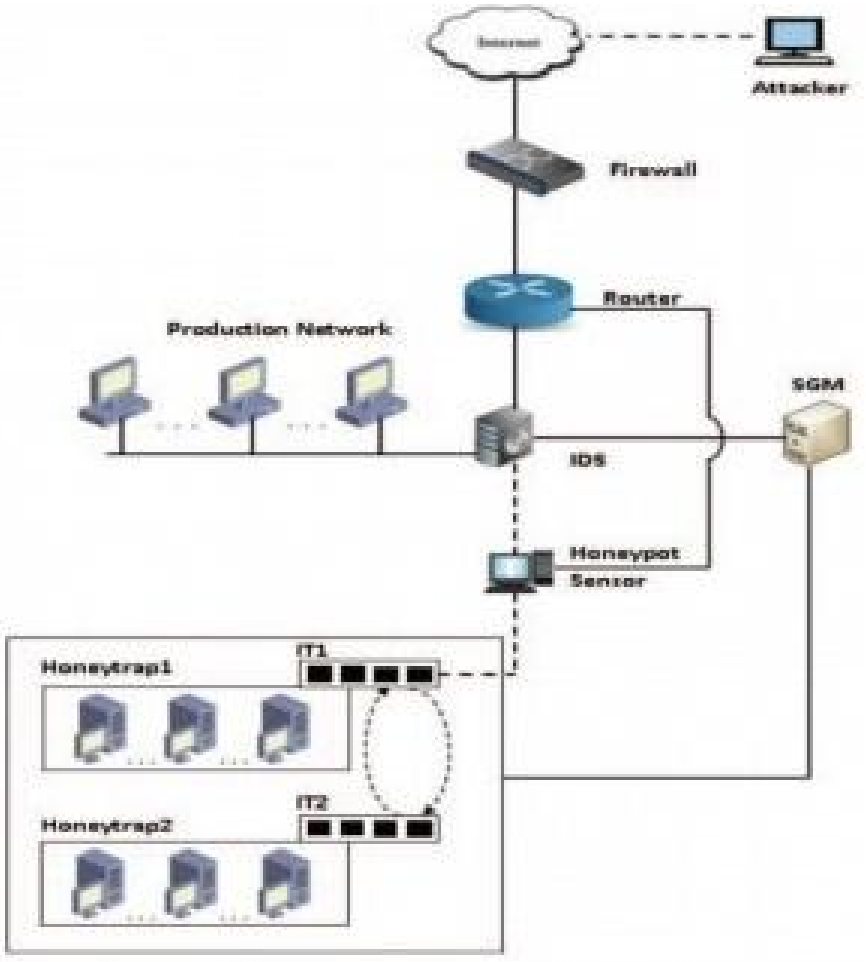


Figure 3.2 System Design of The Glastopf Honeypot Architecture[17]

Honeytrap1 and Honeytrap2 are separate high-interaction honeypots that are multilevel. These are Research Honeypots. Two honeypot traps in a single system capture the attacker so his/her system actions may be recorded and appropriate action taken based on those activities. This approach has three software layers: System, Sebeck, and Application. OS protection and Sebeck Client[18].

1. Data Control: Router-to-honeypot layer bridging isolates honeypot[19]. This bridge allows the attacker in, but he must leave. Since bridge has dual-layer capability, an attacker cannot discover IP address, MAC-address, routing path, etc. Honeytrap develops an outgoing connection to attack. The router's internal translator1 (IT1) sends traffic to honeytrap2[19]. Honeytrap2 tries to link to outside systems like honeytrap1. Honeypot2 transfers traffic to honeytrap1 using internal translator (IT2). This system only enables certain connections since it serves two objectives. First, the system can store enough malicious traffic, thereby reducing DoS attacks[19].

2. Data Capture: In order to analyze malicious traffic correctly and to generate signature from that data, different phases of information would be required and for that single layer would not be so effective and because of that, this module uses multilayer data storing system. Firewall checks the preliminary header information and also filters out any malicious activity. Second layer of this is between router and honeypot sensor. In this multilayer capture system honeypot is the last stage of storing. It stores different kind of data required for generating signature from worms[20].

Honeypot is a system to collect intelligence. Honeypots are usually located behind the firewall. Honeypot mainly used to simulate a variety of services and holes, to induce the occurrence of various attacks, attack data. When an intruder tries to enter the system with a fake identity, the administrator system will be notified. According to Open Web Application Security Project (OWASP) some top attacks recorded were SQL injection and XSS[9] When someone tries to enter the system, a log is generated about all the entries. Even though the intruder succeeds in entering the system and captures the data from the database, the intruder can be fooled by providing fake data, this is done by honeypot, but intruder

will not be aware about this fake information. So, this this can save our system and fool intruders. At the same time the logs will be created, so that all the data about attacker are recorded like system IP, attack type, attack pattern, available footprints etc., and attack method for the evidence which can be used for further action[21].

3.3.1 Conceptual Design

On a real computer running RedHat 7.3 Linux, VMware Workstation with a bridged network connection and settings produced virtual addresses on the network. After installation, the Modern Honey Network –11 platform was setup using default parameters. The honeypot was linked to a firewall's DMZ arm and given a public address[22]. The firewall was designed to prohibit undesired network access, while the IDS complemented the Honeypot's protection and detection. In case of danger, it alerted system administrators[22]. Several tools were installed in the honeypot to identify intruders and report them to safeguard the production system. Honeypot Sensor (HS) detects attackers and reports it to the date module, where the attacker's intelligent information is captured and sent to Activities Downloader (AD)[22]. Fake Honeypot Detection (FHD) is set to swiftly identify any phony honeypot produced by attackers to trick and confuse a genuine honeypot system. The quantity of attackers' intelligent information gathered and the actual honeypot system's rapid reaction to identify fake ones will decide the honeypot's efficiency and efficacy in protecting webmail server information[22].

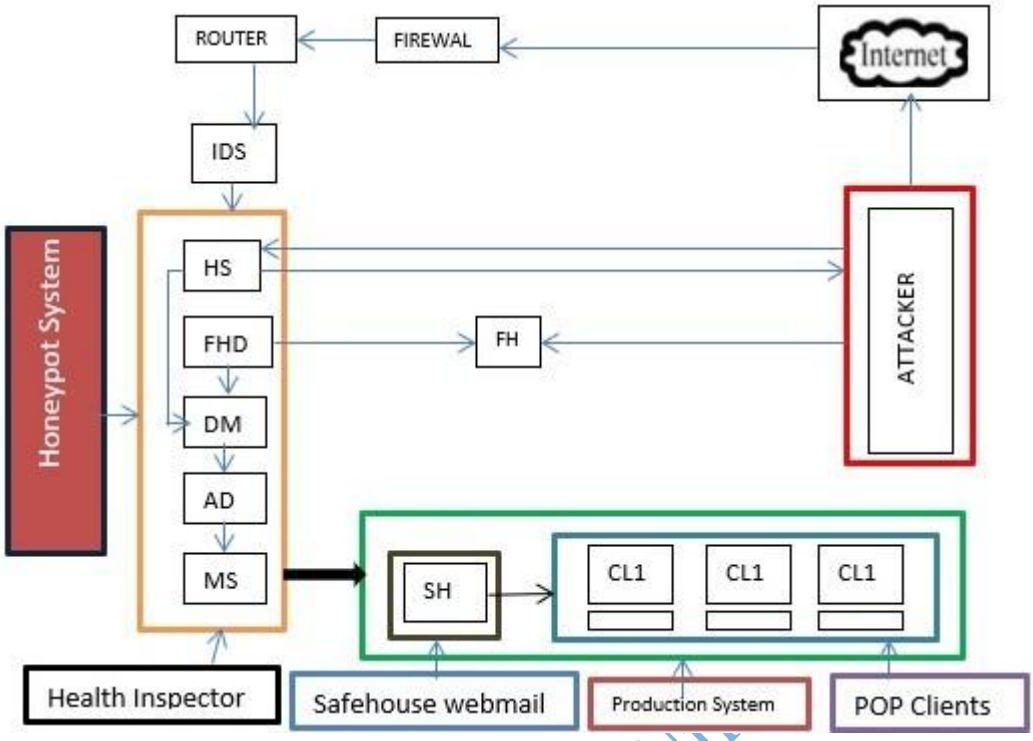


Figure 3.3: Conceptual Diagram of Glastopf *Honeypot system simulation for the detection of Fake honeypot* ***(Researcher: A. B. Owolabi)***

The Glastopf honeypot receives information about the fake honeypot lunched by the attacker on a vulnerable system, with the aim of creating confusion and deceive the system user so that the system can be infected through DDoS of Phishing. The user is unaware that an attacker has taken a lunch of fake honey but Glastopf honeypot detects the fake honeypot through the Fake honeypot detection and raises an alert to Data Module (DM) which provides details about the threat to the Security team.

3.3.2 Process Model

Process models are graphical representations of processes based on notations and standards[23]. Due to the nature of this study, it involves designing, building, and testing a honeypot system to analyze and benchmark its efficacy in acquiring attacker information and safeguarding cloud server documents. This work uses the waterfall concept[23].

3.3.2.1 Waterfall Model

A waterfall model divides project operations into linear sequential phases, where each phase relies on the preceding one's deliverables and specializes tasks[24]. Honeypot development cyber feeds into system design and maintenance[24]. This process paradigm is used owing to its top-down, linear flow of operation from system design through maintenance. This is similar to the architecture of honeypot systems, which must go through the same procedure before being installed to safeguard and secure cloud data[24]. The Waterfall Model is one of the original software development models because it explicitly outlines each stage with logical information flow[24].

The correct model depends on resources and project criteria. Despite its flaws, the Waterfall Model remains popular for small-scale projects. Early knowledge of requirements guides developers throughout the development process. There is no space for frequent adjustments, which makes the model inflexible[24].

Why Waterfall Model?

Since the waterfall model is suitable for the project for the following reasons, it has been decided that the waterfall model will be adopted to complete the project:

1 Time Management: Time can be managed accordingly from the starting point and the researcher will know the time of completion of each phase.

2 Structure: The model is structured with definite phases that allow researchers to have a fixed concentration on the phases.

3 Order: The management of phases is orderly poised.

4 System Development Life Cycle (SDLC): It provides unique deliverables for the project.

5 Visibility: The sequential model can assist the researchers in seeing a downward flow for each phase until each phase ends.

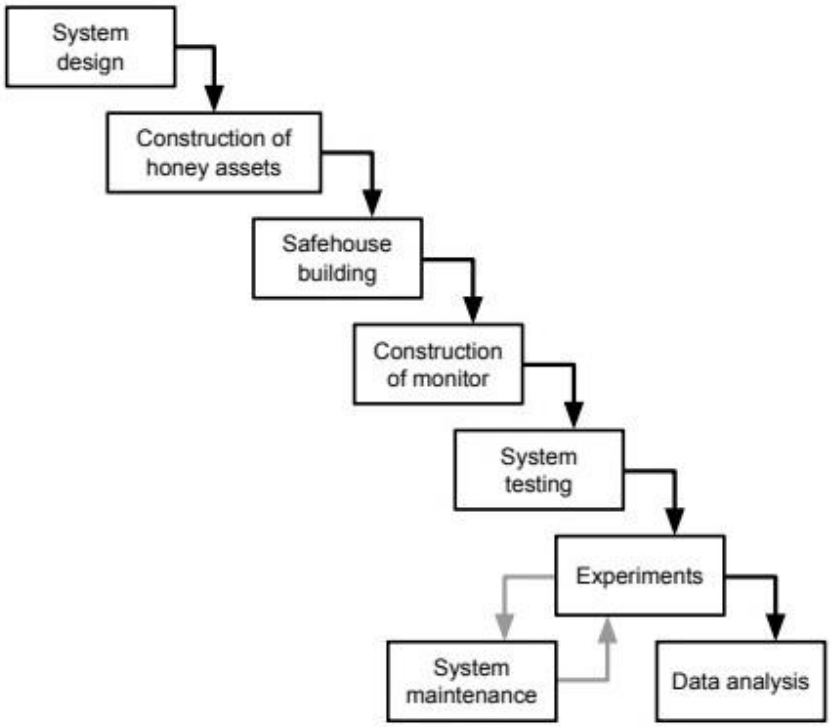


Figure 3.4: Waterfall model (Honeypot development life cycle)[24]

3.4 Data Collection Tool and Techniques

3.4.1 Type of Data Collection

For the purpose of this research work, a secondary data type that is tagged "**hornet40-traffic-perhoneypot-h"** which was generated by **Velerose Veronica** and uploaded to the **Mendeley Data repository** was adopted and used on honeypots system to test and assess the effectiveness of honeypots system on malicious activities via decoy account[25]. The proposed data will assess and test the effectiveness of the honeypots system on cloud-based accounts generated through social media sites like Facebook, Instagram and Twitter, Cloud documents and webmail accounts like email, Gmail, twitter, LinkedIn, Instagram etc. A

simulated honeypots application will be run on each account to test the level of defense that can be created by the honeypots application to guide against Infiltration.

Different social media accounts, webmail accounts, and cloud documents will be assessed to identify the amount of attack and danger on the platform and the degree of damage to courseware platforms from different schools[25].

3.4.2 Description of Data Collection

Hornet 40 is a dataset of 40 days of network traffic assaults acquired on cloud honeypots to study network attack intake and honeypot detection[26]. Data was acquired in April, May, and June 2022 from eight honeypots. The data includes binary and text network flows with 118 features and 480 bytes apiece. Argus collected the data. Passive honeypots lack software hence attacks are passive. Amsterdam, Bangalore, Frankfurt, London, New York, San Francisco, Singapore, and Toronto honeypots collected data. The data covers April, May, and June 2022[26].

3.4.3 Data Source

The information was gathered from eight Digital Ocean[2] honeypot cloud servers. Amsterdam, Bangalore, Frankfurt, London, New York, San Francisco, Singapore, and Toronto are among the cities represented[72].

3.4.4 Data Accessibility

The Hornet 40 dataset is available on Mendeley Data. DOI for the dataset is http://dx.doi.org/10.17632/tcfzkbpw46[27].

3.4.5 Data Format

This collection is referred to as 'raw flow data,' and it contains network flows generated from raw network data. Eight Linux cloud passive honeypot computers collected raw network data.

3.4.6 Data Analysis Tool

Omnet++ is an open-source simulation design management software. It's Unix and Windows-compatible. It automates installation, patch management, and deployments[28].

This analysis used Omnet++ to imitate network honeypots. Pmnet, a component-based Python simulation toolkit and network simulator were utilized. Eclipse supports Omnet++. It adds wizards, editors, and other features to writing simulation[28]. It can mimic wireless and wired networks, protocols, queuing networks, and multiprocessors. Free, unlike OMNEST. This design utilizes OMNET++ to install and automate servers. Virtual IP addresses combine honeypot systems with networks. It's the only cyber deception simulator[28].

3.5 Research Methods

3.5.1 Methods of Achieving Each of the Objectives

1) Objective One: Design a model for performance enhancement and of existing Glastopf honeypot to accurately detect the fake honeypot developed by the attackers

Method

First method is to build a virtual machine with the use of a tool called Omnet++ on a system running on Linus Operating System. Omnet++ is a component-based C++ simulation library and framework. The simulation tool was picked owing to its specific intrinsic capabilities that make it more useful and suitable in designing a network and web

based cyber security architecture[29]. It's one of the most popular simulation tools for IT specialists in cyber security deception technologies. The back end of the virtual system runs Linux, while the front end runs Windows. The **Glastopf Honeypot** system was configured on a virtual simulation system with numerous functional API tools to enhance its capabilities to attract cyber criminals[29].

Although there is no specific API provided for Glastopf optimization purposes, however, there are various tools and techniques that are designed to optimize the performance of Glastopf honeypot. This also provide configuration file where customization of various settings related to its behavior, performance and parameters modifications such as the number of allowed simultaneous connections, thread pool size, logging options, etc can be carried and adjust these settings according to specific requirements. Web Server Optimizations tools were also configured to emulate the vulnerable web applications. This was leveraged on the APIs and configuration options provided by the web server (such as Apache or Nginx) to optimize its performance, this includes options like enabling caching, adjusting worker processes or threads, configuring connection timeouts, and fine-tuning request handling. Log Analysis Tools was also configured to analyze logs generated by Glastopf which helps in providing insights into its performance and any potential performance bottlenecks. Utilize log analysis tools like ELK Stack (Elasticsearch, Logstash, and Kibana) or Splunk to collect, parse, and visualize Glastopf logs. These tools can help identify patterns, analyze traffic, and detect performance issues. Other API configuration tools that were installed on Glastopf honeypot are Load Testing Tools such as Apache JMeter or Siege were used to simulate heavy traffic and assess the performance of the Glastopf honeypot under different load conditions. By analyzing the results, one can

identify performance bottlenecks and take appropriate optimization measures. Performance Monitoring Tools such as Nagios, Zabbix, or Prometheus were also configured to continuously monitor the resource usage, network metrics, and other performance parameters of Glastopf honeypot. These tools provide real-time insights and can help in identifying and resolving any performance issues. It is important to note that optimizing Glastopf involves a combination of configuration changes, utilization of external tools, and monitoring. The specific optimizations will depend on the environment, network setup, and requirements.

a. Network Settings

Honeypot requires a network adaptor. Honeypots are allocated static IP addresses inside the same segment. VMware Workstation's eth0 interface is setup with a virtual switch for the Honeynet's internal network[32].

b. Time Synchronization

It's setup logs timestamps match and data may be evaluated. Network Time Protocol does this (NTP).

c. Creating Users

Creating a Glastopf honeypot user in a web server and deliberately assign weak password and direct traffic to the web applications to make it vulnerable and attracted to the attackers

d. Installation of Additional API Tool for Fake Honeypot Detection

The API tool was installed and setup on Glastopf honeypot virtual machine to identify any form of fake honeypot that could be deployed by the attackers, the API tool was deployed to determine the level of performance evaluation of enhanced Glastopf honeypot in detecting fake honeypot and its ability to gather attackers' intelligent information. Below

are the additional API tools that were installed and configured for the performance optimization of Glastopf honeypot.

e. Installing and Configuring MySQL

MySQL is setup for network accessibility. Keep the setting default. Following initialization, the database server may be started[34].

f. Installing and Configuring Sebek

In the honeypot, Sebek's source code is downloaded and built. It is open source that operate like path code, this is necessary in order to assist the API to detect any strange application on the network and quickly raise alert to the system administrator[35].

g. Configuring IP Tables

By default, the firewall allows all Honeypot traffic. All the IP address will be recorded on IP table for proper monitoring and filtering. In the honeywall's IP Tables, rules will be placed to govern the traffic to the Honeynet[36].

h. Honeywall **Roo Configuration**

Honeynet needs a honeywall to generate a virtual computer. The honeywall has three interfaces: the administrative interface, the external and internal Honeynet interfaces[37].

Honeywall Roo produces a GUI. eth0 and eth1 are bridge interfaces. IP-less interfaces. Honeywall and its bridge interface are hard to detect. Honeywall Roo's rc. firewall script configures the bridge interface and firewall rules[37].

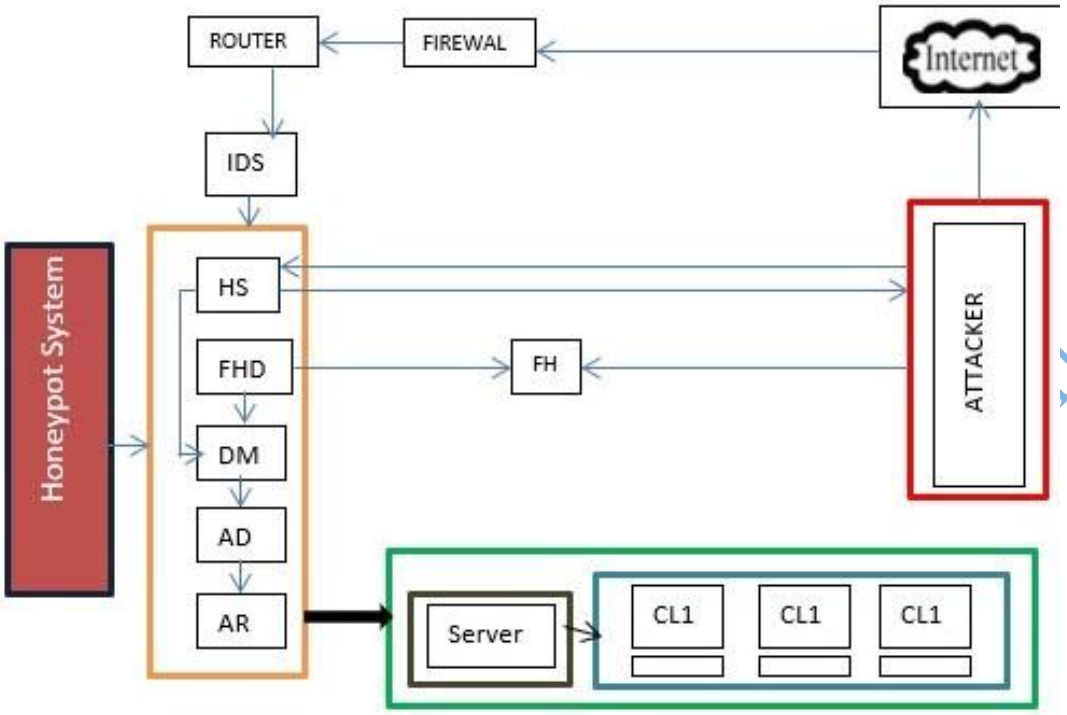


Figure 3.5: Conceptual Diagram for Objective One ***(Researcher: A. B. Owolabi)***

Key:

IDS: Intrusion detection system
HS: Honeypot Sensor
FH: Fake Honeypot
FHD: Fake Honeypot Detector
DM: Data Module
AD: Activities Downloader
RM: Reactor Module
CS: Cloud Server
CL: Clients

Expected Result

Build and implement a powerful, robust and operational enhanced Glastopf honeypot system on Linux Redhat for accurate collection of attackers' intelligent information and detection of fake honeypot system makes easier with the configurations of the necessary API tools.

2) Objective Two: Implement the designed model on a Web Application to determine fake honeypot system and gathering of attackers' information

Method

The implementation of Glastopf Honeypot was carried out using a virtual Laboratory experiment through which a web server was set up. Recall that Glastopf honeypot system was defined as a web-based Low Interaction honeypot designed to emulate vulnerable web applications to attract intruders. However, a Glastopf honeypot-running on virtual machine has web server where web applications are designed and made vulnerable to attract attackers. The first step that was taken during the implementations was choosing a Linus operating system for deployment and installed all the required dependencies like Python and Apache HTTP server. An open source Glastopf from the official GitHub repository or using package managers like pip. was downloaded and installed. The configuration of Glastopf was done to match the environment by modifying the configuration files and set the desired logging level, network settings, and honeypot behavior. There was identification and specification of web pages applications that were emulated for vulnerability. Some realistic web applications were launched to accommodate the generated the data collected from Venerose through Mendeley data repository with known vulnerabilities using frameworks like PHP, Java, or .NET. Framework after which it customizes the web server's response to match vulnerable conditions and deployed the emulated web applications within Glastopf. The network configuration settings were carried out to redirect incoming traffic to Glastopf honeypot. This was done by setting up port forwarding, using network filtering tools like IP tables, or placing Glastopf behind a dedicated network monitoring device. The honeypot logs were monitored and analyze the

data captured from potential attacks on Venerose web application, and this was done by utilizing tools like log analyzers and SIEM solutions to identify attack patterns and detect any potential threats. There was setting up a mechanism for regular updates with the latest releases of Glastopf and vulnerability databases. This is to regularly updateed Glastopf and emulate web applications to keep pace with new attack vectors and vulnerabilities. Glastopf honeypot as intergrated with other security tools and systems, such as Intrusion Detection Systems (IDS) and Security Information and Event Management (SIEM) solutions for sharing of data and alerts between Glastopf and these systems in order to enhance the overall security posture. The system was designed to continuously monitor the honeypot's performance, ensuring stability and efficient resource utilization and carry out regularly review and analyze honeypot logs to identify new attack techniques or emerging threats. It is necessary for periodically review and update honeypot configurations to adapt to evolving attack scenarios and to improve the effectiveness of the honeypot. At the end of the implementation, the number of attackers logged on web applications and fake honeypots detected at various deployment levels web applications would determine the level of efficiency of Glastopf honeypot and its future in this challenging cybercrime environment[39].

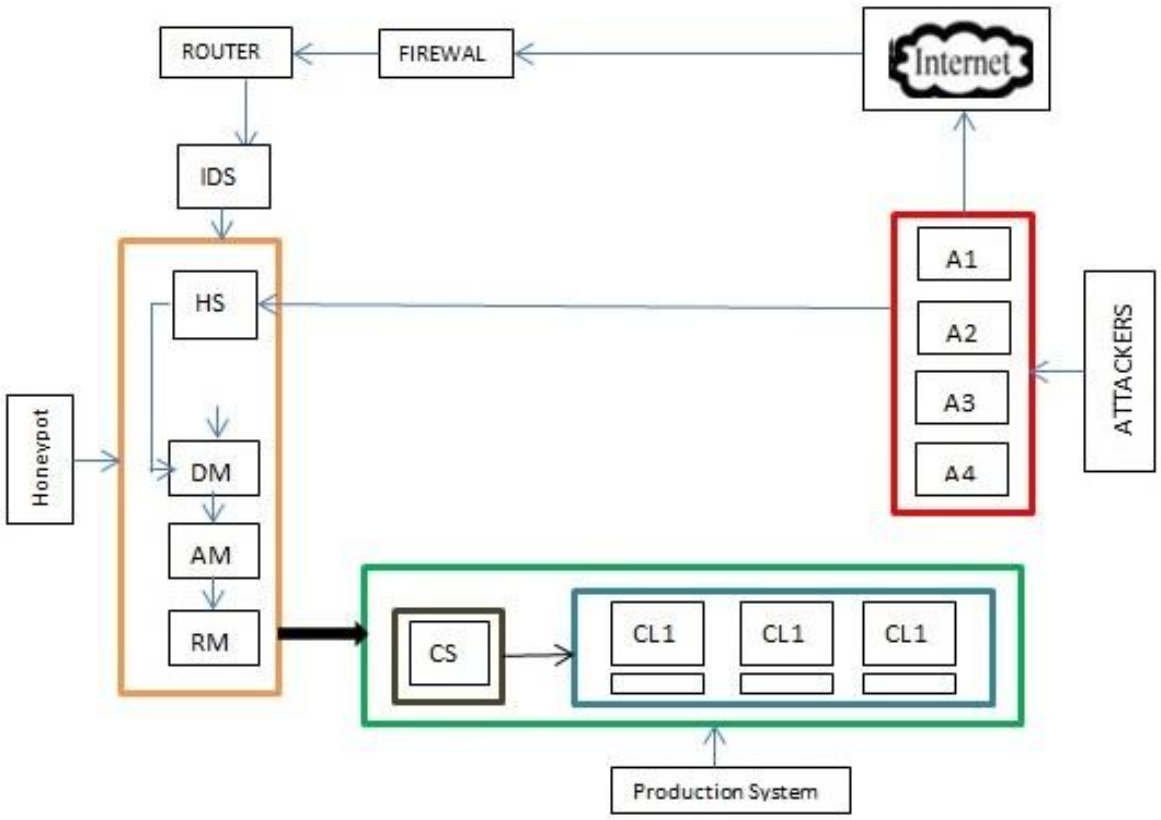


Figure 3.6: Conceptual Diagram for objective two on Implementation of honeypot on Webmail service ***(Researcher: A. B. Owolabi)***

Expected Result

It is expected that this method of Glastopf honeypot implementation on vulnerable web application would provide better security defense on web server and its information by detecting fake honeypot system and gather attackers' intelligent information for investigation and forensic analysis, this would also improve the level of performance of the honeypot system.

3) Objective Three: Test the level effectiveness of the enhanced Glastopf honeypot system in detecting fake honeypot designed by the attackers

Method

A Metric-based technique which include Trap efficiency, Attack Diversity, Attack Frequency, Session Duration, False Positive rates, Attacks origin Analysis, Field Testing, Logs Analysis and Comparative Testing, were used to test the level of efficiency of the enhanced Glastopf honeypot in terms of attacker's intelligent information gathering from the overall number of attacks on venerable web applications.

Trap efficiency: In order to determine the level of efficiency of Glastopf honeypot, the trap efficiency metric testing was used to measure how effective the enhanced Glastopf honeypot is at attracting and capturing malicious activity from the attackers. It can This was done by comparing the number of interactions with the honeypot system to the number of interactions with legitimate web server.

Attack diversity: This was carried out to measure the variety of attacks that the Glastopf honeypot is able to attract. It was done by analyzing the different types of attacks detected by the honeypot, such as SQL injection, cross-site scripting, or remote code execution. The number of the attacks that Glastopf honeypot can attack based on different attack mentioned would determine the level of efficiency of the honeypot in term of performance.

Attack frequency and Session Duration This metric was used to measure the rate at which attacks are attempted on the honeypot. It was calculated by analyzing the number of attack attempts over a specific period of time. While the Session Duration metric measures the length of time that attackers spend interacting with the honeypot which was calculated by analyzing the duration of each session recorded by the honeypot.

Attack origin analysis and False Positive Rtae: This metric measures the geographic and network location of the attackers targeting the honeypot. It was calculated by analyzing the IP addresses or network patterns associated. While False Positive Rate metric measures the rate at which the honeypot incorrectly identifies legitimate activity as malicious and was calculated by analyzing the number of false positives generated by the honeypot compared to the total number of events captured.

Field Testing: This was used to deploy the Glastopf honeypot in a real-world environment, such as a production network or a simulated network with realistic user behavior. It was used to monitor the honeypot over an extended period of time, collect data on attack attempts, payload captures, and detection rates. This method provides insights into real-world effectiveness but may present challenges in controlling variables. The analysis on logs generated by the honeypot to gain insight into detected attacks and their characteristics were carried out. This helps to determine the level of the effectiveness of Glastopf honeypot especially on gathering attackers' information and detection of fake honeypot system. The assessment on the captured payloads, extracted malware, or malicious scripts to understand the nature of the attacks faced by the honeypot were also caried out. This method provides valuable information about the types and sophistication of attacks. Another metric test that was carried out was comparison testing. This was based on the deployment of multiple honeypots, including enhanced Glastopf, alongside each other and monitor the honeypots simultaneously and compare their performance in terms of attack capture rates, detection accuracy, and false positive/negative rates. This method helps determine how Glastopf stands against other honeypot solutions.

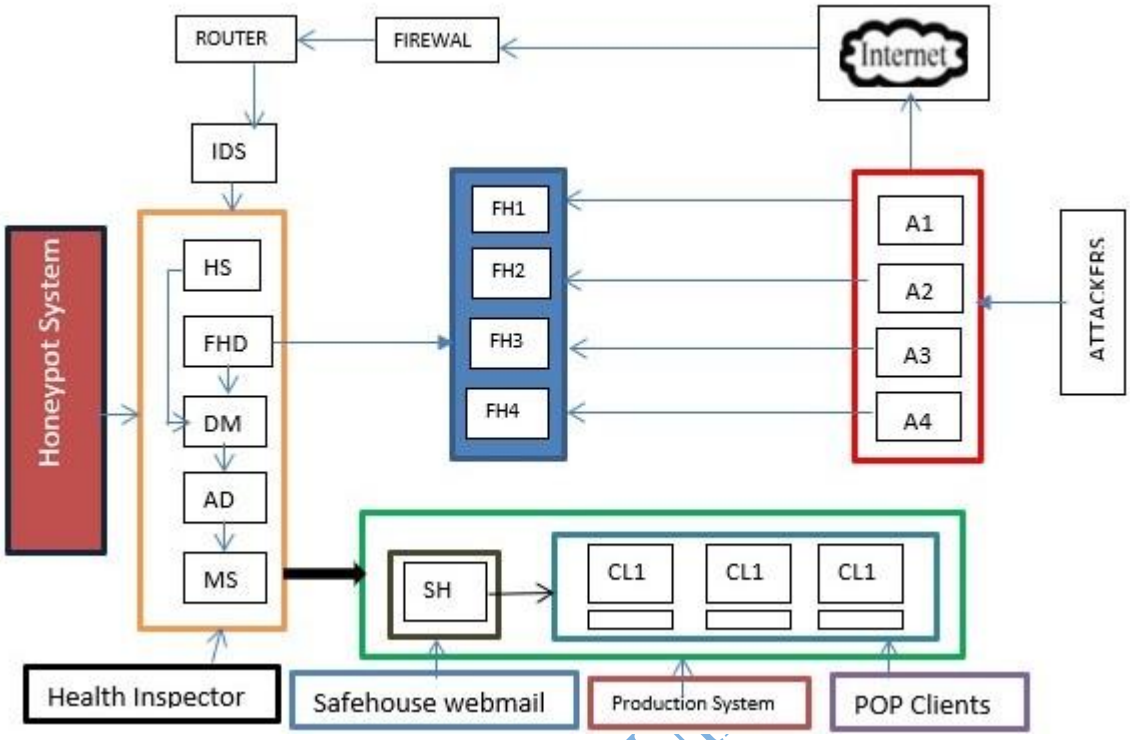


Figure 3.7: Conceptual Diagram for objective three on testing of efficiency of honeypot on Webmail service ***(Researcher: A. B. Owolabi)***

Expected Result

It is expected that enhanced Glastopf honeypot should be able to capture attackers' intelligence from sensor data, safeguard the data in the cloud, and identify attackers' phony honeypot systems and detect fake honeypot system.

4) Objective Four: Evaluate the performance of enhanced Glastopf in (iii) against the existing honeypot system in the detection of fake honeypot system and attacker's information gathering

Method

To achieve this objective, there are indices that are required to compare the effectiveness and performance of existing Glastopf honeypot with enhanced Glastopf in terms of gathering attackers' information and detection of fake honeypot. The result of enhanced

Glastopf honeypot in objective (iii) was compared and correlated with the result of existing honeypot system. The assessment results were visually displayed using bar, pie, and line charts[44]. The analysis of each of the below performance evaluation metrics would assist greatly in the evaluation of enhanced Glastopf's performance in different aspects like gathering attackers' intelligent information and detection of fake honeypot. and its operations compared with other honeypots. The performance evaluation indices are listed below which are also tabulated: its ability to handle traffic, resource efficiency, effectiveness in capturing attacks, reliability of logging, and ease of use. It is important to balance these metrics based on your specific requirements and environment.

Table 3.1 Metrics for Performance Evaluation

Performance Evaluation Metrics	Description
Traffic Handling	Measure the honeypot capability to handle web traffics
Resource Utilizations	Assess the usage of memory, CPU and disk Resources
Event logging	Evaluate the Quality and Quantity of generated logs
Attack detection	Determine the honeypot ability to detect and capture attacks
False Positive	Assess the rate of False alarm and noise in the logs
Ease of deployment and Configuration	Evaluate the usability and ease of setting up Glastopf
Community Support and Updates	Consider the availability of support and regular update

Expected Result

The newly designed enhanced Glastopf honeypot system is expected to perform better than the existing one in terms of intelligent information gathering and fake honeypot detection due to the inclusion of a fake honeypot detection tool and honeypot sensor in collaboration with IDS that can sense every incoming attack and report it to the main honey through information translator[45].

3.6 Ethical Consideration

Ethical considerations guide research designs and techniques. To comply with professional ethics in thesis writing, the following area of research ethics was addressed and taken care of:

1• Informed permission: As part of ethical research consideration in order not to infringe on the right of other researchers, an official letter was sent to Velerose for permission to use her secondary data for the purpose of research. This is in accordance with the extant law that guide against the exploration of someone else intellectual property.

2• Participation voluntarily: All the research scholars that made contribution to the research study did that voluntarily when their consent for participation were sought.

3• No harm: During the period of implementation, all the data collected from velerose were carefully integrated to the system in such a way that important and confidential information would not be affected.

4• Confidentiality: A confidential agreement was signed with the administrator of virtual cloud where virtual attacks were lunched. There was an assurance that none of the vital information of the cloud server would be divulged to the public of for enterprise usage.

All of the above ethical confirmation in research writing was considered, and none of it was dismissed.

3.7 Method of Result Dissemination

Research results might be published in local and international journals, conference proceedings, academic bulletins, and national newspapers. Honeypot technology will be made accessible to business and financial organizations as a deceitful tool for protecting cyber information[48].

3.8 Target Audience

This study targets public and commercial businesses who secure their data using virtual cloud servers. Others include social media, webmail, and small, medium, and big corporate network managers.

Endnotes

[1]A. Abdulrahman, M. Ishaq, A. Fatima, A. Atika & Y. Suberu. "*A Proposed Improved Captcha Based Intrusion Detection Model*"**, Journal of Advanced Science and Optimization Research** Vol. 27, No.9, 2023 ISSN 2418-9325

[2]A. Ahmim, L. Maglaras, M. Ferrag, M. Derdour & H. Janicke. "*A Novel Hierarchical Intrusion Detection System Based on Decision Trees and Rules-based Models*". In 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), 2019, 228-233, DOI: 10.1109/DCOSS.2019.00059

[3]A. Arkhipova & D. Karevskiy. *"Honeypot as a Tool for Creating an Effective Secure System".* Novosibirsk State Technical University in Digital Technology Security Digital Technology Security 2021 ; https://doi.org/10.17212/2782-2230-2021-2-122-135

[4]A. Christin, C. Giselle, A. Wesam, A. Abu & S. Maha. "*A Comprehensive Survey for IoT Security Datasets Taxonomy, Classification and Machine Learning Mechanisms.* **Computers & Security,** 2023, 103283, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2023.103283.
(https://www.sciencedirect.com/science/article/pii/S0167404823001931)

[5]A. Elkosairy & A. Marianne. "*A New Web Deception System Framework"*, Conference: 2018 1st International Conference on Computer Applications & Information Security (ICCAIS) 2018, DOI: 10.1109/CAIS.8442027,

[6]A. Mishra & Sanjay K. Jain. "*A Survey on Question Answering Systems with Classification*". **Journal of King Saud University-Computer and Information Sciences** 28.3 2016, pp. 345–361. https://doi.org/10.1016/j.jksuci.2014.10.007

[7]A. Mudgal & S. Bhatia. **"***Spark-Based Network Security Honeypot System: Detailed Performance Analysis***"** Article, Dec 2022, **International Journal of Safety and Security Engineering.** 12. 2022, 737-743. 10.18280/ijsse.120610.

[8]A. Pashaei, Mohammad E. Akbari, Mina Z. Lighvan & C. Asghar. *"Early Intrusion Detection System using Honeypot for Industrial Control Networks"*, Results in Engineering, Volume 16, 2022, 100576, ISSN 2590-1230, https://doi.org/10.1016/j.rineng.2022.100576.

[9]A. Riancho. W3AF USER GUIDE. Available at: URL: http://cyber. lockheedmartin. com/hubfs/Gaining the Advantage Cyber Kill Chain. 26, 2021.

A. Schlenker, O. Thakoor, H. Xu, F. Fang, M. Tambe, L. Tran-Thanh, P. Vayanos & Y. Vorobeychik. "*Deceiving Cyber Adversaries: A Game Theoretic Approach* 17th International Conference on Autonomous Agents and Multiagent Systems, AAMAS Volume: 2, 2018 pp. 892–900.

[10]A. Shah. "*Evaluating Network Forensics Applying Advanced Tools*". **International Journal of Advanced Engineering, Management and Science**, Vol 9 No 4 2023, http://journal-repository.theshillonga.com/index.php/ijaems/article/view/6178

[11]A. Waqas, A. Muhammad, N. Sabreena & W. Farhana. "*Detection and Analysis of Active Attacks using Honeypot*". **International Journal of Computer Applications** (0975 – 8887) Volume 184 – No. 50, 2023 IJCATM: www.ijcaonline.org

[12]A. Yaser. "*Improving Intrusion Detection Systems Using Artificial Neural Networks*". **ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal**, Vol. 7 No. 1 2018 https://doi.org/10.14201/ADCAIJ2018714965

[13]B. Gupta & A. Gupta. "*Assessment of Honeypots: Issues, Challenges and Future Directions*". **International Journal of Cloud Applications and Computing (IJCAC)** 8(1) 2018 |Pp 34 DOI: 10.4018/IJCAC.2018010102

[14]B. Mphago, & S. Mpoeleng. "*Deception in Web Application Honeypots: Case of Glastopf*". **International Journal of Cyber-Security and Digital Forensics**, 6(4), 2017, pp. 179-185, DOI: 10.17781/P002304

[15]B. Paul & M. Rao. "*Zero-Trust Model for Smart Manufacturing Industry*". **Applied Sciences Journal.** 13(1) 2023 :221. https://doi.org/10.3390/app13010221

[16]B. Sara, C. Mauro, P. Luca & P. Pier. "*Social Honeypot for Humans: Luring People through Self-Managed Instagram Pages*". **Journal of Social and Information Networks, cs.SI), Artificial Intelligence (cs.AI), Cryptography and Security (cs.CR)** 2023 https://doi.org/10.48550/arXiv.2303.17946

[17]B. Temmie, V. Andrew, J. Kimberly, W. Ferguson, B. Sara, F. Daniel. & E. Kristin. "*The Moonraker Study: An Experimental Evaluation of Host-Based Deception*". In Hawaii International Conference on System Sciences (HICSS), Maui, Hawaii, 2020, DOI:10.24251/HICSS.2020.231.

[18]B. Abbaschian, S. Daniel & A. Elmaghraby. "*Deep Learning Techniques for Speech Emotion Recognition, from Databases to Models*", Computer Science and Engineering Department, University of Louisville, Louisville, KY 40292, USA, 2021, 21(4), 1249; https://doi.org/10.3390/s21041249

[19]C. Chou, C. Wu, K. Lu, L. Hsien & J. Li. *"Modbus Packet Analysis and Attack Mode for SCADA System*" **Journal of ICT, Design, Engineering and Technological Science**. 2018, 30-35. 10.33150/JITDETS-2.2.1.

[20]C. Kai, W. Zhan, L. Dongkun & R. Mu. *"The TaintDroid Based Honeypot Monitoring System for Embedded Device"*, **Journal of Physics Conference Series** 2203 (1):012077, 2022, DOI: 10.1088/1742-6596/2203/1/012077.

[21]C. Kai, W. Zhan, Z. Chengcheng & M. Haohua. *"The Research on Network Function Virtualization Based Network Honeypot"*, Proceedings of the 12th International Conference on Computer Engineering and Networks, 2022, DOI: 10.1007/978-981-19-6901-0_156,

[22]C. Kuan, L. I-Hsien & J. Li. *"Honeypot System of SCADA Security Survey",* Proceedings of International Conference on Artificial Life and Robotics 23:2018 pp 444-447, DOI: 10.5954/ICAROB.2018.OS8-8

[23]C. Sakama, M. Caminada, & A. Herzig. "*A Formal Account of Dishonesty*," **Logic Journal of IGPL,** vol. 23, 2015, no. 2, pp. 259–294, https://doi.org/10.1093/jigpal/jzu043

[24]D. Akshat, B. Anchit, A. Nihal & D. Sumithra. *"HONEYPOT: Intrusion Detection System"* **International Journal of Education Science Technology and Engineering** 3(1): 2020 pp 13-18, DOI: 10.36079/lamintang.ijeste-0301.66

[25]D. Danilov, T. Ovasapyan, D. Ivanov, A. Konoplev & D. Moskvin. *"Generation of Synthetic Data for Honeypot Systems Using Deep Learning Methods"*, Automatic Control and Computer Sciences, 2023, 56(8):916-926, DOI: 10.3103/S014641162208003X

[26]D. Rajesh, Thariq M. Hussan, B. Sri. Vastav. *"Network Protection Using Honeypots"*, **International Journal of Innovative Technology and Exploring Engineering** (IJITEE), Volume-9 Issue-6, 2020 ISSN: 2278-3075 (Online)

[27]D. Velasco & G. Rodriguez. *"A Review Of The Current State Of Honeynet Architectures And Tools"*, **International Journal of Security and Networks** 2017, pp 255-272, DOI: 10.1504/IJSN.10009165

[28]D. Zhang, F. Gang, S. Yang & S. Dipti. "*Physical Safety and Cyber Security Analysis of Multi-Agent Systems: A Survey of Recent Advances*." **IEEE/CAA Journal of Automatica** Sinica 8, no. 2, 2021: 319-333, DOI: 10.1109/JAS.2021.1003820

[29]D. Zielinski & Hisham A. Kholidy . *"An Analysis of Honeypots and their Impact as a Cyber Deception Tactic"*, 2022, DOI: 10.48550/arXiv.2301.00045.

[30]David P. Fidler. "*Just & Unjust War, Uses of Force & Coercion: An Ethical Inquiry with Cyber Illustrations*", Daedalus Vol. 145, No. 4, 2016, pp. 37-49 https://**www.jstor.org/stable**/24916782

[31]E. Abiodu, J. Aman & I Abiodu. "*A Comprehensive Review of Honey Encryption Scheme"*. **Indonesian Journal of Electrical Engineering and Computer Science**, Vol. 13, No. 2, 2019, pp. 649~656 ISSN: 2502-4752, DOI: 10.11591/ijeecs. v13.i2.

[32]E. Fujisaki. "*All-But-Many Encryption*", **Journal of Cryptology,** 31 2018**,** 31**,** pages 226–275, https://doi.org/10.1007/s00145-017-9256-x

[33]E. Iasiello. "*What is the Role of Cyber Operations in Information Warfare*"? **Journal of Strategic Security,** vol. 14, No 4, 2021, pp. 72-86, https://www.jstor.org/stable/48633489

[34]E. Morales, C. Rubio & A. Doupé. "*HoneyPLC: A Next-Generation Honeypot for Industrial Control System***s**", CCS '20: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security 2020, Pages 279–291https://doi.org/10.1145/3372297.3423356

[35]E. Zavadskii. & D. Ivanov. *"Counteracting Information Threats Using Honeypot Systems Based on a Graph of Potential Attacks"*, Automatic Control and Computer Sciences 56(8): 2023, 964-969, DOI: 10.3103/S0146411622080260

[36]E. Zavadskii.& D. Ivanov. *"Implementation of Honeypot Systems Based on the Potential Attack Graph"*, Automatic Control and Computer Sciences, 55(8):2021, 1194-1200, DOI: 10.3103/S0146411621080460,

[37]Eirini A. Anthi, W. Lowri, R. Matilda R, B. Pete, & W. Adam. "*Adversarial Attacks on Machine Learning Cybersecurity Defences in Industrial Control Systems*." **Journal of Information Security and Applications** 58 2021: 102717, https://doi.org/10.1016/j.jisa.2020.102717

[38]F. Kimberly, F. Sunny, M. Justin, & M. Maxine. "*Game Theory for Adaptive Defensive Cyber-Deception*". Proceedings of the 6th Annual Symposium on Hot Topics in the Science of Security 2019 4 Pp 1–8 https://doi.org/10.1145/3314058.3314063

[39]F. McKee & D. Noever. *"Chatbots in a Honeypot World"*, Cryptography and Security (cs.CR); Computers and Society (cs.CY); Machine Learning (cs. LG), 2023, https://doi.org/10.48550/arXiv.2301.03771

[40]F. Wenjun, D. Zhihui, F. David & V. Victor. "*Enabling an Anatomic View to Investigate Honeypot Systems: A Survey*". **IEEE Systems Journal** Volume: 12, Issue: 4, 2017, DOI: 10.1109/JSYST.2017.2762161

[41]Fransiska S. Mukti & Muhammad R. Sukmawan. *"Integration of Low Interaction Honeypot and ELK Stack as Attack Detection Systems on Servers"* **Jurnal Penelitian Pos dan informatika** 11(1), 2021, DOI: 10.17933/jppi.v11i1.336, License CC BY-NC-SA 4.0

[42]G. Anand, T. Neha & H. Nayankumar. "*An Overview Of Honeypot Systems*, **International Journal Of Computer Sciences And Engineering,** 7(2): 2019, pp 394-397, Doi: 10.26438/Ijcse/V7i2.394397

[43]G. Elisavet, L. Athanasios, Panagiotis R. Grammatikis & Panagiotis G. Sarigiannidis. *"Protecting IEC 60870-5-104 ICS/SCADA Systems with Honeypots"*, IEEE International Conference on Cyber Security and Resilience (CSR), 2022, DOI: 10.1109/CSR54599.2022.9850329

[44]G. Tsochev, M. Sharabov & A. Georgiev. "*Using Machine Learning Reacted with Honeypot Systems for Securing Network*", International Conference Automatics and Informatics (ICAI), 2021, DOI: 10.1109/ICAI52893.2021.9639590

[45]H. Chie. "*Using The Modified Diffie-Hellman Problem to Enhance Client Computational Performance in A Three-Party Authenticated Key Agreement*". Arab. Journal. Of Science. Engineering. 43 (2), 2018, pp 637–644. https://doi.org/10.1007/s13369-017-2725-6

[46]H. Muhammad, Olumide B. Longe & Adebisi A. Baale. "*Towards the Development of a Machine Learning Enhanceed Framework for Honeypot and CAPTCHA Intrusion Detection Systems*" **Advances in Multidisciplinary and scientific Research Journal Publication** 2022, DOI: 10.22624/AIMS/ACCRABESPOKE2022/V34P4,

[47]H. Yuan, X. Changyou, D. Ke, Z. Guomin & S. Lihua. "*A Differential Privacy Based Multi-Stage Network Fingerprinting Deception Game Method*", **Journal of Information Security and Applications**, Volume 74, 2023, 103460, ISSN 2214-2126, https://doi.org/10.1016/j.jisa.2023.103460.(https://www.sciencedirect.com/science/article/pii/S2214212623000443)

[48]H. Zhou, C. Dong, R Wu, X. Xu & Z. Guo. "*Feature Fusion Based on Bayesian Decision Theory for Radar Deception Jamming Recognition"*, IEEE Access, vol. 9, 2021, pp. 16296-16304, doi: 10.1109/ACCESS.2021.3052506.

Chapter Four

Results and Discussion of Findings

This section includes the analysis of the attack patterns observed on the honeypots and presents an overall statistical analysis of the results gathered using the high interaction honeypot. The analysis was based on conducting a laboratory experiment and testing of honeypot on hijacked webmail account, stolen social account and cloud documents to measure and access the level of efficiency and effectiveness of honeypot technology to prevent and safeguard the accounts. This will help in determining the level of efficiency and ability of honeypot system to detect fake honeypot being deployed and executed by the attackers to deceive and cause confusion for the security administrator. The research focused on a particular honeypot system called Glastopf was focused on and a tool for the enhancement of it was developed for its operation for effective gathering of attackers intelligence information and also detecting the fake honeypot system that could be developed by the attackers to deceive and confuse the real honeypot system.

4.1 Attacks that are Supported by Glastopf

There was identification of the attacks that are supported and can be handled by the Glastopf honeypot. Glastopf is a low-interaction honeypot, and it has been designed for this project based on fulfilling a purpose. It supports limited functionalities and can be extended if the development is required by the administrator. Glastopf can support the following attacks:

Table 4.1 Supported attacks by Glastopf **(Researcher: A. B. Owolabi)**

S/N	Attack Type	Supported by Glastopf
1	SQL Injection	Yes
2	Cross Site Scripting (XSS)	Yes
3	Command Injection	Yes
4	Remote File Inclusion	Yes
5	Local File Inclusion	Yes
6	Sensitive Date Explosure	Yes
7	Directory Traversal	Yes

During the experiment, the aforementioned table portrays the attacks that can be supported by Glastopf and how the web application is designed as well. However, the list possesses common attacks that can be formulated by the attackers through simulating the web-based honeypot and the activities will surely be logged in the honeypot system.

4.2 Results and Discussions for Objective One: Design a model that can enhance the performance of existing Glastopf honeypot to accurately detect the fake honeypot developed by the attackers

4.2.1 Design for the Deployment of Glastopf Honeypot

This chapter explains the design for the deployment of Glastopf honeypot. It also emphasized the security break down simulation that could be done in the next chapter. In addition, it is also illustrated how the system design would be in the implementation part of the research

4.2.2 Why Glastopf Honeypot?

One of the facilities of the honeypot is being an open-source software[1]. It suggests that it is the output of extraordinary brainstorming of the experts and the software will have continuous improvement as time passes by.

Cross-site scripting, web application, and database can significantly expose attack which might have exploitation and then it can send spams, can be able to convert the website to the bots, and lastly, can serve the attacks. Glastopf is regarded as a "low-interaction honeypot" that can emulate different web servers that host numerous web pages as well as web applications whereas all of these possess vulnerabilities. Glastopf is not very difficult to settle in and once it is indexed by the search engines, the attacks will be poured in numbers daily. Glastopf is run with the Python web application which is low-interaction and works as a network emulator. One of the interesting attributes of the tool is the emulation capacity since the application can be vulnerable to the SQLi attacks[2].

4.2.3 Web Based Honeypot Design

Glastopf web application honeypot is different than other honeypots available in the market as it has the capability to interact with different web-based attack for instance XSS, Command injection, XHTML injection, RFI. SQL injection can also be emulated by Glastopf since it is a low-interaction web application honeypot[3]. The honeypot was installed in the ubuntu server in VMware. The honeypot can be operated on the python scripting, and it can be activated with the PHP sandbox. In the Linux server, the PHP sandbox is installed. The capability of Glastopf let this having all interactions that can take place with this attacker by storing the logs. Though the Glastopf honeypot is a low

interaction honeypot, it has the potential to serve the administrator with a great degree of security. The Glastopf can be served by the website and can be utilized through the initiation of python and can be represented as a form of a PHP application to the attacker[4]

a. IP Address of the attacker and the victims are as follows:

Attacker: 192.168.50.155

Victim: 192.168.50.162

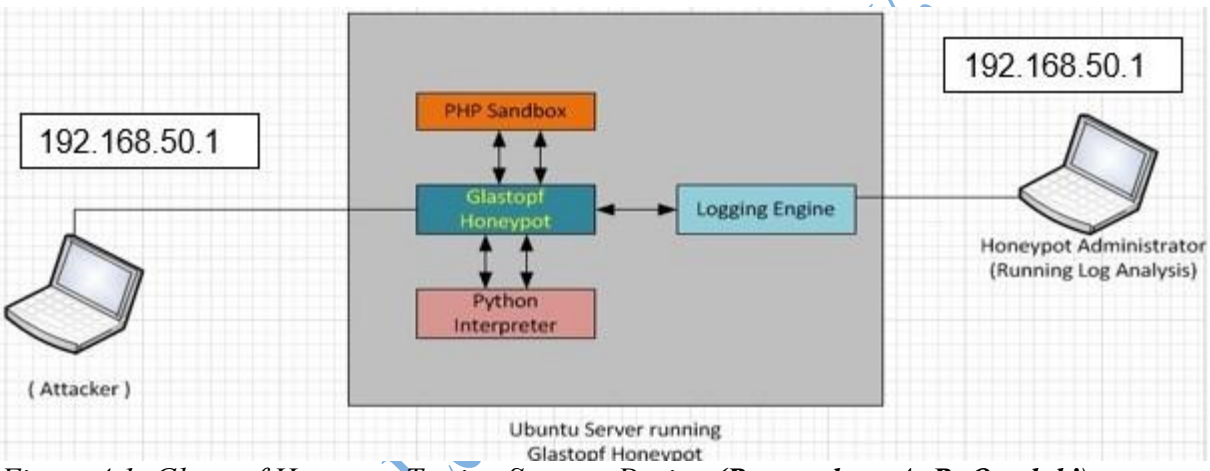


Figure 4.1: Glastopf Honeypot Testing Systems Design ***(Researcher: A. B. Owolabi)***

The figure mentioned above illustrates the Glastopf system, which is needed to be tested. Then the honeypot will experience the probe, and the open-source application will be used to attack the system. The open-source security tool should be equipped with the backtrack Linux. The honey will run the harmful code (e.g., PHP codes) which will already be given by the attackers. The operation was taken place in the sandbox, and therefore the result was directly sent to the attackers. That means the attackers will be able to see what they wanted to have, such as the output of the attacks when the attacks bring success for them. Besides, the security tools would make sure the simulation of different attacks which are given on the honey pot system. The system administrator is responsible for the analysis of the logs

which are generated while the attack is going on. This level is regarded as one of the approaches that can be followed so that the 'test system' for the Glastopf can be created.

4.2.4 The Simulation of the Web Attack in the Honeypot-Controlled Environment:

Numerous attacks can be simulated in the web application honey pots. It is necessary to mention that every attack can easily be replicated, and the replication will be done using different third-party software and tools. Upon the requirement, the manual testing procedure can also take place so that the honey pot systems can be tested. The simulation of the attacks can be conducted while the experimental stage is going on.

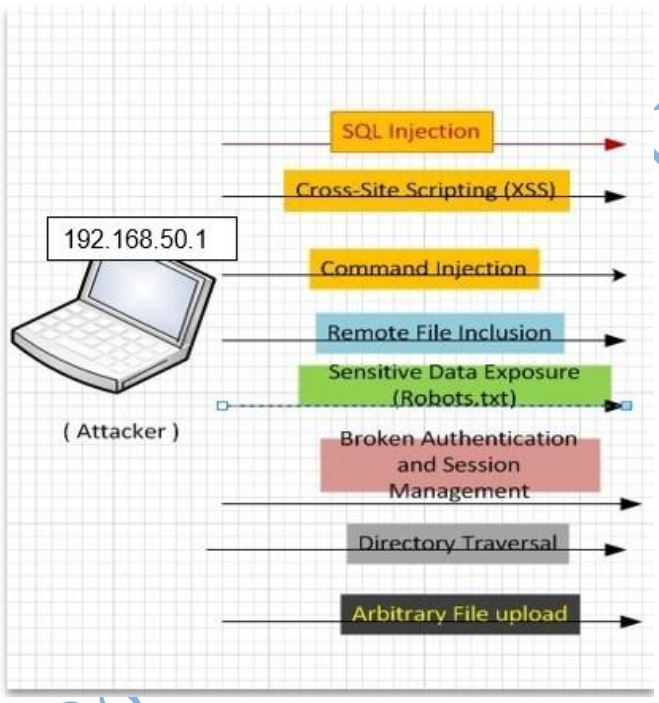


Figure 4.2: Breakdown of the Security Tests ***(Researcher: A. B. Owolabi)***

The aforementioned figure expresses the breakdowns of simulations which are needed to be conducted in the honeypot web application so that the analysis can be done while the process is still going on.

4.2.5 Simulated Attack Design

The following Figure 4.3 depicts the high-intensity attack in the Glastopf honeypot system.

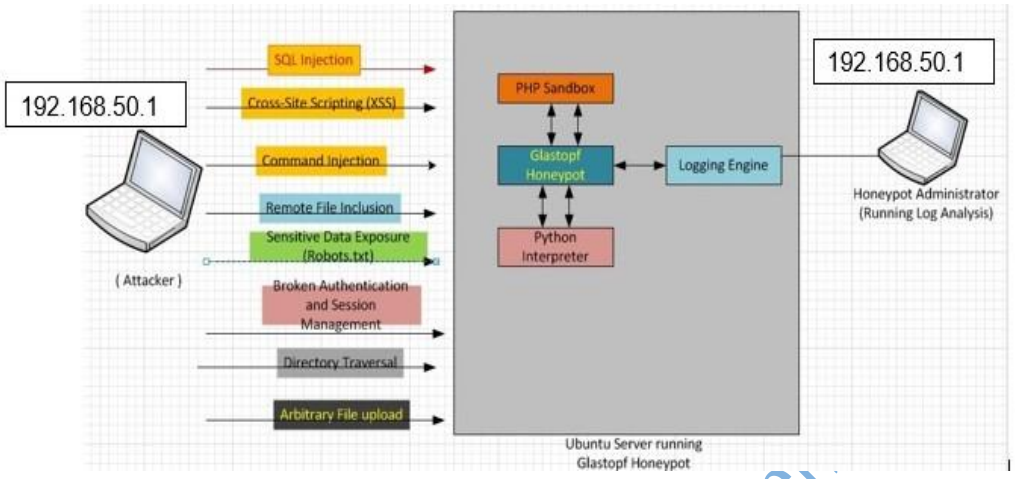


Figure 4.3: Test of Glastopf Honeypot ***(Researcher: A. B. Owolabi)***

The system in the run time can capture numerous logs since the attacks can have simulation by using different testing software and tools. It can be expected from the honey pot so that the user and all the associated data of the user can be captured that is combined within the Glastopf honey pot web application system. Afterwards, the logs can be used so that the analysis can be done on different patterns of the attacks, and output can be achieved after the experiment of the honeypot.

4.2.6 Activities

A simulator was designed with the use of OMNETT++ simulation tool to serve as a virtual system where honeypot system was installed and configured to monitor activity in webmail account information towards understanding malicious activity in compromised system. Other interested researchers may install their own cloud document accounts for similar studies using the system's credentials and codes. In the course of this research, it was discovered that this is the second publicly available cloud document honeypot

infrastructure that will be set up to allow external researchers to make contributions and create accounts on webmail, social media account and cloud document information that is configured on a simulated system. The accounts created are residing in a simulated system and do not have any connection with the real account but rather for research purposes. This design focuses on two areas: The production system has the Cloud server and distant clients, whereas the honeypot system is meant to trick and draw attackers.

Several cloud-based accounts, including Facebook, Gmail, and cloud document accounts, were established as dummies in the production system and leaked their credentials via underground forums, public paste sites, and malware-infected virtual computers. As honey on the honeypot, false webmail, social networking, and cloud document accounts were established to attract attackers. The detailed measurements of the activity logged by the honey accounts were provided and documented over a period of 15 days.

The wealth of information individuals saves in webmail accounts like Gmail, Yahoo! Mail, or Outlook, and the risk of losing them for unauthorized acts, fascinate hackers who compromise such accounts[5]. These hackers get victims' account credentials by phishing, infecting users with information-stealing software, or hacking big password databases, betting on the fact that online account users commonly reuse passwords[37]. Cybercriminals may utilize stolen account information and data covertly or convert it to corporate data and sell it[5]. Cybercriminals exploit hacked accounts in numerous ways. It's often used to send spam[6]. This strategy is successful because account owners trust known connections and open their messages[6]. Since the stolen account has a history of good behaviour with the hosting provider, dangerous information delivered from it may not be detected as spam, especially if the recipients are within the same webmail service (e.g., a Gmail account that

spams other Gmail accounts)[6]. Cybercriminals take data from stolen accounts. This includes financial data, student portal logins, and victim information[6].

4.3 Results and Discussions for Objective Two: Implement the designed model on a Web Application to determine fake honeypot system and gathering of attackers' information

The **Glastopf** honeypots was configured and deployed in the Linux servers for the purpose of the experiment. The honeypots system has accessibility to the web application, and every deployment is implemented in the virtual machines that was designed with OMNET++ simulator. Therefore, the hacker will find it identical and it would be regarded and acted like an actual attack done by a hacker. When the hacker will see the web application and he / she will prob to attack the website. The experiment is done based on necessarily stood upon the designed approach, and it has been mentioned in the aforementioned section.

4.3.1 Honeypot Configuration on OMNET++ Simulation Tool

After designing a platform using OMNET++ on Ubuntu where honeypot system configuration can be carried out, various inherent tools were installed to allow the system to acquire intelligent information about attackers and identify the likelihood of attackers designing bogus honeypot systems. Apache module was setup to attract attackers using puppet module install puppetlabs-apache, followed by MYSQL server module configuration file. SMTP and FTP servers were also established for webmail data delivery.

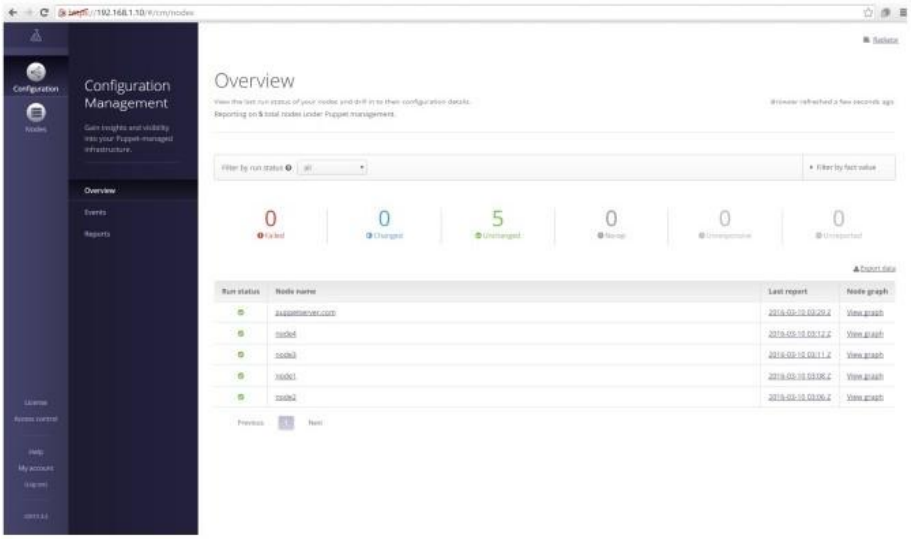


Figure 4.4: Overview Configuration of the Honeypots ***(Researcher: A. B. Owolabi)***

4.3.2 Tools Installed in the Ubuntu for the Configuration and Implementation of Honeypot System on Glastopf

The first tool that was installed and configured on Unbutu running on OMNET++ virtual machine was Glastopf honeypot. Different commands that were used in the configuration and deployment have been mentioned. Configuration and installation of the Glastopf is also illustrated in this section. All the tools and operating system that were used in the attack process were also mentioned. The following section also portrayed different types of experiments that had been done in the system by attacking the deployed honeypot system[7].

4.3.2.1 HonSSH Configurations on Honeypot for the Detection of Fake Honeypot

HonSSH is an interactive Honeypot. Static, it lies between an attacker and honeypot, creating two SSH connections. Scripted, it creates too many SSH connections between an attacker and honeypots[8]. HonSSH sits between the attacker and Puppet honeypots. Bifrozt has HonSSH. HonSSH can detect a fake honeypot and alert the data module. False

honeypots are detected. The following command will let the OpenSSH server to install in the ubuntu machine[8].

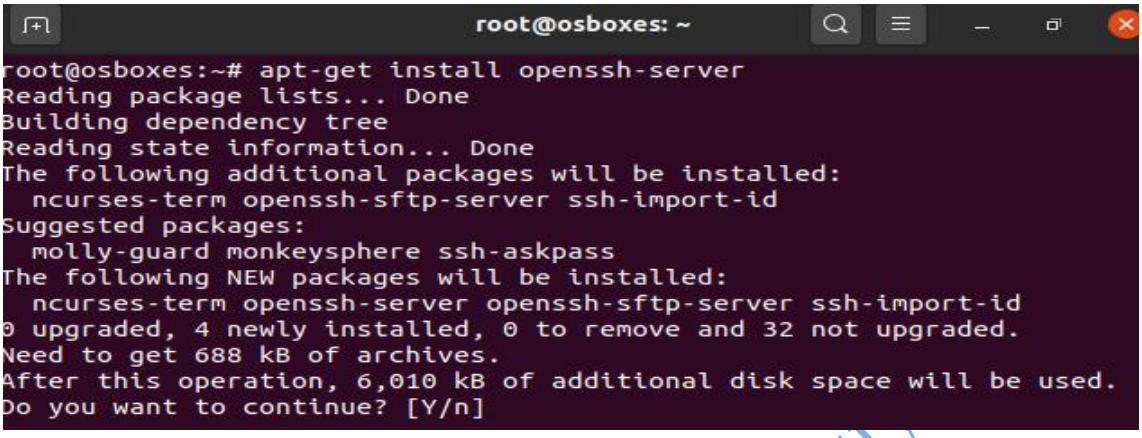


Figure 4.5: OpenSSH Server has been Installed in The System

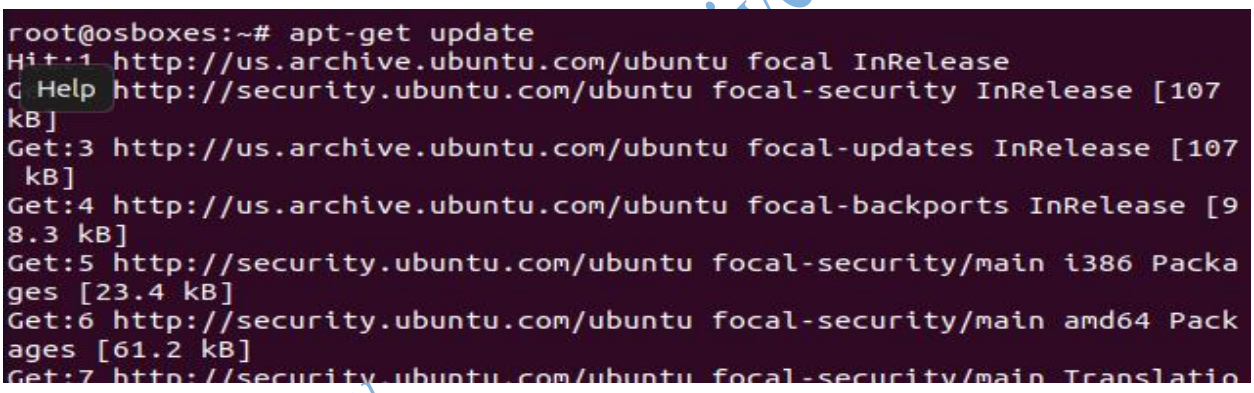


Figure 4.6: Updating the System

After OpenSSH server been installed, the upgradation of the system needed.

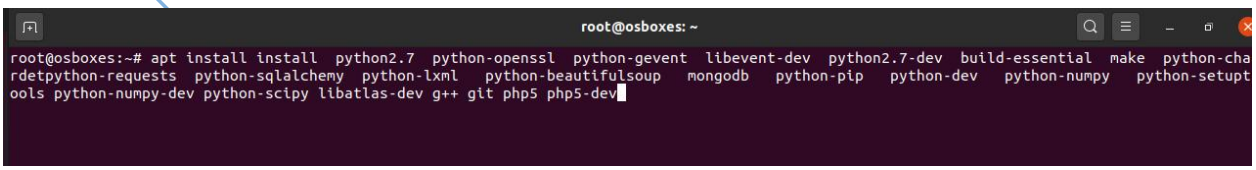


Figure4.7: All Dependencies be Installed

Whilst installing the Glastopf, it is required to install the python dependencies and OpenSSH library. There are some other libraries that are also required like php5 development libraries and C/C++ compiling libraries. After installation of the dependencies, it is required to update and upgrade the machine.

```
oot@osboxes:~# apt install libatlas-base-dev
eading package lists... Done
uilding dependency tree
eading state information... Done
he following additional packages will be installed:
  libatlas3-base libgfortran5 libquadmath0
uggested packages:
  libatlas-doc liblapack-doc
he following NEW packages will be installed:
  libatlas-base-dev libatlas3-base libgfortran5 libquadmath0
 upgraded, 4 newly installed, 0 to remove and 105 not upgraded.
eed to get 8,682 kB of archives.
fter this operation, 51.9 MB of additional disk space will be used.
o you want to continue? [Y/n]
```

Figure 4.8: Ubuntu Requires Libatlas-base-dev to be Installed

The newer version of Ubuntu requires libatlas-base-dev installed as well.

```
root@osboxes: ~
root@osboxes:~# apt install python3-pip
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu build-essential dpkg-dev fakeroot g+
+ g++-9 gcc gcc-9 libalgorithm-diff-perl
  libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan5 libatomic1 libbinutils libc-
dev-bin libc6-dev libcrypt-dev libctf-nobfd0
  libctf0 libexpat1-dev libfakeroot libgcc-9-dev libitm1 liblsan0 libpython3-dev libpytho
n3.8 libpython3.8-dev libpython3.8-minimal
  libpython3.8-stdlib libstdc++-9-dev libtsan0 libubsan1 linux-libc-dev make manpages-dev
 python-pip-whl python3-dev python3-distutils
  python3-setuptools python3-wheel python3.8 python3.8-dev python3.8-minimal zlib1g-dev
Suggested packages:
  binutils-doc debian-keyring g++-multilib g++-9-multilib gcc-9-doc gcc-multilib autoconf
 automake libtool flex bison gcc-doc
  gcc-9-multilib gcc-9-locales glibc-doc libstdc++-9-doc make-doc python-setuptools-doc p
ython3.8-venv python3.8-doc binfmt-support
The following NEW packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu build-essential dpkg-dev fakeroot g+
+ g++-9 gcc gcc-9 libalgorithm-diff-perl
```

Figure 4.9: Install PIP3 Installed

Pip needs to be installed, if pip3 is not installed.

```
                                Pythons.
root@osboxes:~# apt-get update
Get:1 http://security.ubuntu.com/ubuntu focal-security InRelease [107 kB]
Hit:2 http://us.archive.ubuntu.com/ubuntu focal InRelease
Get:3 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease [107 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease [98.3 kB]
Fetched 312 kB in 1s (232 kB/s)
Reading package lists... 73%
```

Figure 4.10: Update the Distro

Update the distro again after installing all the required libraries and software's.

After the installation of the dependencies next step is to install the sandbox which is running on Php application under sandbox environment.

```
root@osboxes:/opt# apt install git
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-el git-email
  git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 100 not upgraded.
Need to get 5,464 kB of archives.
After this operation, 38.4 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Figure 4.11: GIT Installed

Git is also needed to install further sandboxing tools and Environment.

```
root@osboxes:/opt# git clone git://github.com/glastopf/BFR.git
Cloning into 'BFR'...
remote: Enumerating objects: 57, done.
remote: Total 57 (delta 0), reused 0 (delta 0), pack-reused 57
Receiving objects: 100% (57/57), 15.33 KiB | 178.00 KiB/s, done.
Resolving deltas: 100% (26/26), done.
root@osboxes:/opt#
```

Figure 4.12: Cloned the BFR in Opt Directory

Cloned the BFR in opt directory.

```
root@osboxes:/opt# apt install php7.4-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  autoconf automake autopoint autotools-dev debhelper dh-autoreconf
  dh-strip-nondeterminism dwz gettext intltool-debian
  libarchive-cpio-perl libarchive-zip-perl libcroco3
  libdebhelper-perl libfile-stripnondeterminism-perl libltdl-dev
  libmail-sendmail-perl libpcre2-16-0 libpcre2-dev libpcre2-posix2
  libsigsegv2 libssl-dev libsub-override-perl
  libsys-hostname-long-perl libtool m4 php-common php-pear php-xml
  php7.4-cli php7.4-common php7.4-json php7.4-opcache
  php7.4-readline php7.4-xml pkg-php-tools po-debconf shtool
Suggested packages:
  autoconf-archive gnu-standards autoconf-doc dh-make gettext-doc
  libasprintf-dev libgettextpo-dev libtool-doc libssl-doc gfortran
```

Figure 4.13: Developing the Sandboxing Environment

phpize needed to be installed. This command is used for the development of the sandboxing environment of php extension.

```
root@osboxes:/opt/BFR# phpize
Configuring for:
PHP Api Version:         20190902
Zend Module Api No:      20190902
Zend Extension Api No:   320190902
root@osboxes:/opt/BFR#
```

Figure 4.14: Showing the Zend Module Size and API

Phpize shows Zend module API no and extension no.

```
root@osboxes:/opt/BFR# ./configure --enable-bfr
checking for grep that handles long lines and -e... /usr/bin/grep
checking for egrep... /usr/bin/grep -E
checking for a sed that does not truncate output... /usr/bin/sed
checking for pkg-config... /usr/bin/pkg-config
checking pkg-config is at least version 0.9.0... yes
checking for cc... cc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
```

Figure 4.15: Configuration Enabling

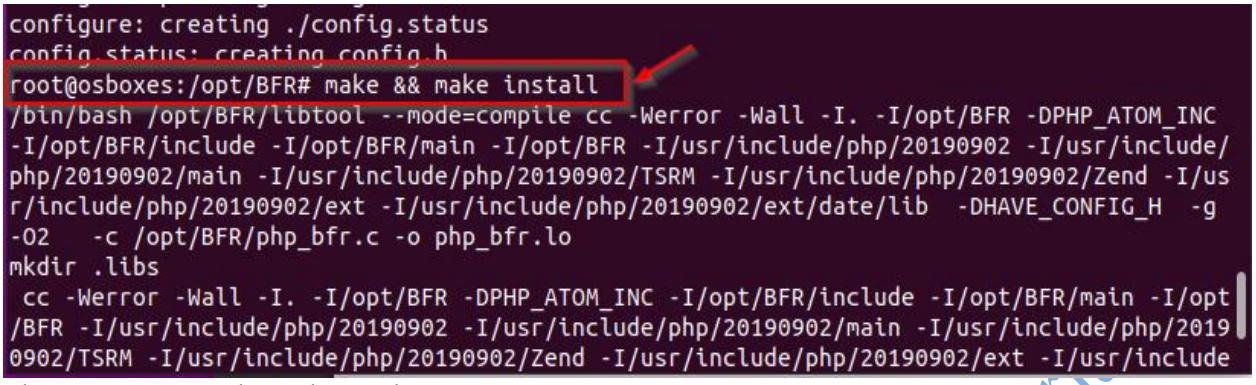


Figure 4.16: Creating Directories

After configuration it is needed to get the link from the lib directory.

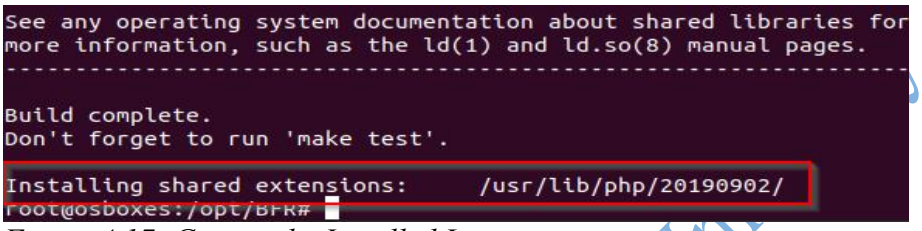


Figure 4.17: Getting the Installed Location

The php sandbox installs in the below mentioned location and added with the zend extension

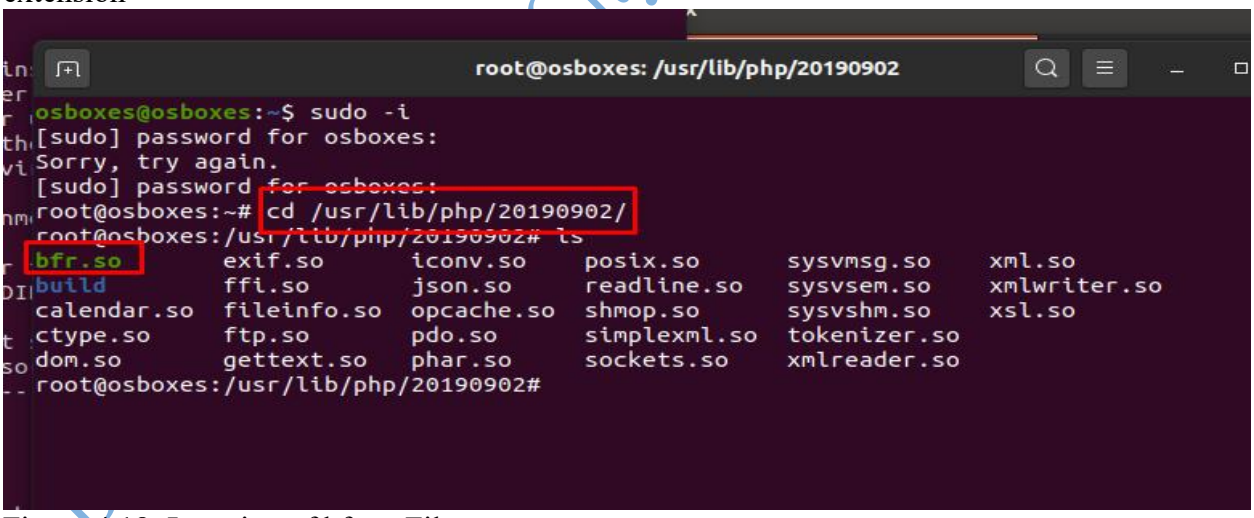


Figure 4.18: Location of bfr.so File

The location of the bfr.so file is located there.

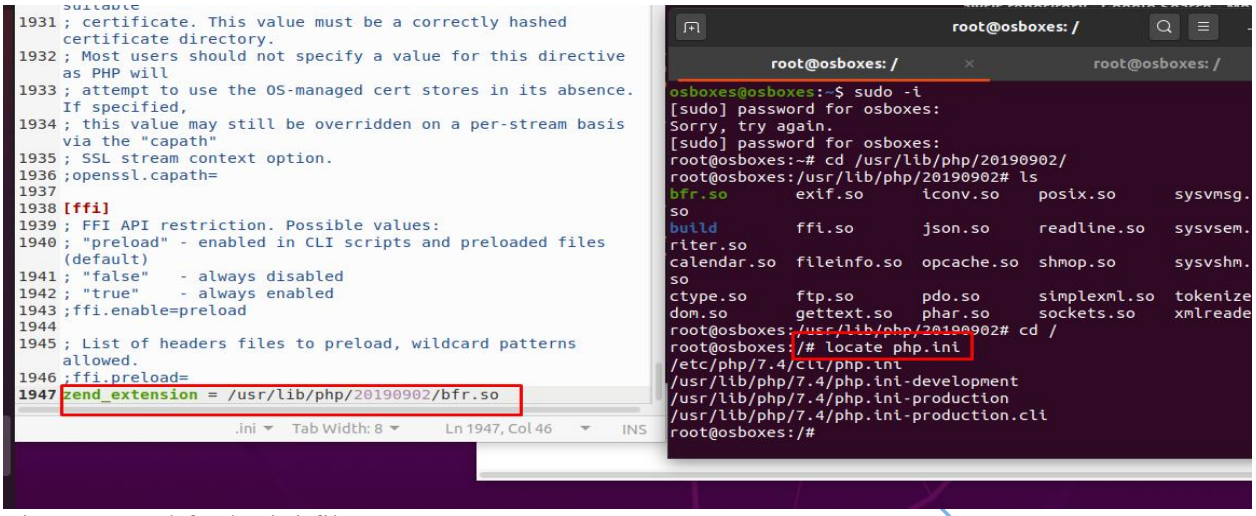


Figure 4.19: bfr php.ini file

The highlighted command has been used to generate the php engine extension, it is needed to update the php.ini file in the /etc/php/7.4/cli/ location.

From the git repository the honeypot is downloaded and installed, the following are the command to install and download the file from git.

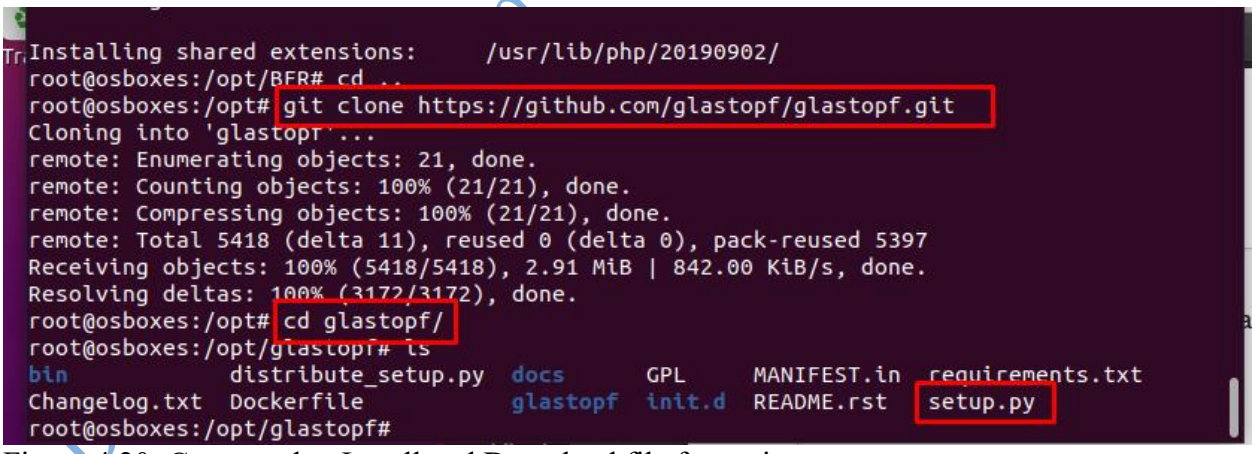


Figure 4.20: Command to Install and Download file from git

Firstly, git cloned the Glastopf and then cd to the folder. Finally, the installation process starts by the following command.

```
  command 'python' from deb python-is-python3

root@osboxes:/opt/glastopf# python3 setup.py install
/usr/lib/python3/dist-packages/setuptools/dist.py:473: UserWarning: Normalizing '3.1.3-de
v' to '3.1.3.dev0'
  warnings.warn(
running install
running bdist_egg
running egg_info
creating Glastopf.egg-info
writing Glastopf.egg-info/PKG-INFO
writing dependency_links to Glastopf.egg-info/dependency_links.txt
```

Figure 4.21: Location of the Glastopf Honeypot

The location of the Glastopf honeypot is:

```
/usr/local/bin/glastopf-runner
```

In the opt directory the folder has been created earlier to install Glastopf and where Glastopf.cfg file can be found

```
 GNU nano 2.2.6                File: glastopf.cfg

#If disabled a sqlite database will be created (db/glastopf.db)
#to be used as dork storage.
enabled = True
#mongodb or sqlalchemy connection string, ex:
#mongodb://localhost:27017/glastopf
#mongodb://james:bond@localhost:27017/glastopf
#mysql://james:bond@somehost.com/glastopf
connection_string = sqlite:///db/glastopf.db

[surfcertids]
enabled = False
host = localhost
port = 5432
user =
password =
database = idsserver

[syslog]
enabled = False

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

Figure 4.22: Configuration of the Glastopf

This is the last configuration file of the Glastopf honeypot system. The final step of the installation process is to generate the configuration file on the Glastopf environment. There is a bash script that can run of pop up the honeypot.

The Glastopf honeypot runs tcp port 80 by default in all the interfaces. If someone is testing the IP address, will find a website with bunch of service running.

If a web scanner like Nikto or Acuentix is run on this IP, the attacker will find some vulnerabilities and try to attack. 4.2.2.2 Configuration of the Logging Procedure in the Honeypot:

The server was configured in such a way that the logs of the server is been redirect to the separate file in /<installed dir>/glastopf.cfg:

```
[logging]
consolelog_enabled = True
 filelog_enabled = True
logfile = log/glastop.log
```

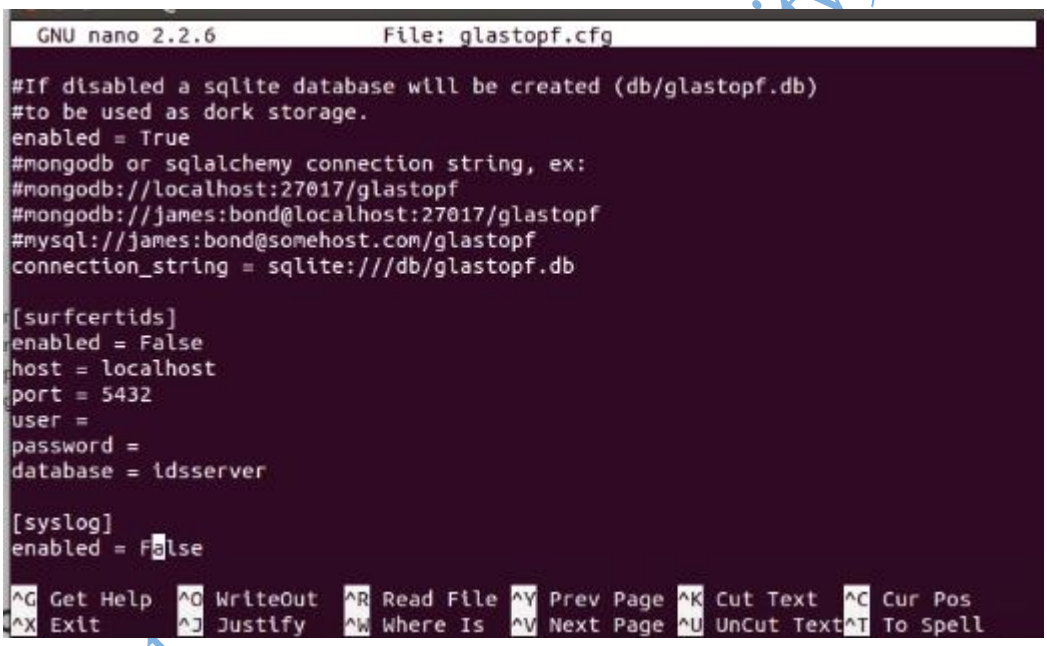


Figure 4.23: Glastopf.cfg File

The server also been configured to redirect the logs to different sql database in the different directories.

```
[main database]
#If disabled a sqlite database will be created (db/glastopf.db) to be used
as dork #storage.
enabled = True connection_string = sqlite:///db/glastopf.db
```

It is designed in such an order so that the errors generated by the honeypot will be redirected to a separate log file in location as below and the database logs will be redirected to the separate sqlite file in another separate file in different location [/<installed dir>/db/glastopf.db]

```
/<install-directory>/log/glastopf.log
```

The aforementioned log file, as well as the db file, can readily be analyzed through querying the sqlite db file by adopting SQLite3 utility or even it can be analyzed manually.

(It is recommended to install "sudo apt-get install sqlite3 libsqlite3-dev" package

4.3.3 Elasticsearch, Logstash, and Kibana Configuration

The below figure 4.2.3 shows ELK/Logstash and Bifrozt. Elasticsearch 2.2, Logstash 2.2, and Kibana 4.4 were installed on Ubuntu 14.04 Server (ELK/Logstash Server) to aggregate and display honeypot logs. Logstash parses and saves logs. Kibana can search Logstash logs. Elasticsearch stores logs for both programs. (2015) Anicas.

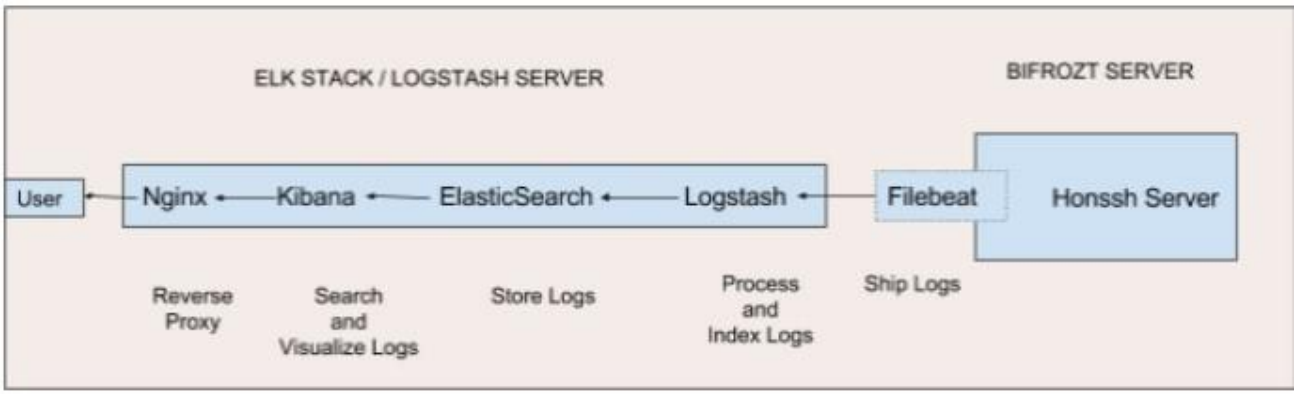


Figure 4.24: ELK Stack Server and Bifrozt Server

4.3.4 Install Java 8

1. use the command sudo add-apt-repository -y ppa:webupd8team/java

2. sudo apt-get update

3. sudo apt-get -y install oracle-java8-installer

4.3.5 Install Elasticsearch

1. wget -qO - https://packages.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add –

2. echo "deb http://packages.elastic.co/elasticsearch/2.x/debian stable main"

3. sudo apt-get update

4. use the command sudo apt-get -y to install elasticsearch

5. Use the command sudo vi /etc/elasticsearch/elasticsearch.yml

localhost is the value for network.host in

7. restart elasticsearch service using sudo

8. sudo update-rc.d elasticsearch defaults 95 10

4.3.6 Install Kibana

1. echo "deb http://packages.elastic.co/kibana/4.4/debian stable main" | sudo tee -a /etc/apt/sources.list.d/kibana-4.4.x.list

2. sudo apt-get update

3. sudo apt-get -y install kibana

4. sudo vi /opt/kibana/config/kibana.yml server. host is set to "localhost"

6. sudo update-rc.d kibana defaults 96 9

7. sudo service kibana start

4.3.7 Install Nginx

Because Kibana was setup to listen on localhost, there was setting up a reverse proxy to enable access from the Internet. Nginx was used for this.

1. run the command sudo apt-get install nginx apache2-utils

2. sudo htpasswd -c /etc/nginx/htpasswd.users kibanaadmin

3. sudo vi /etc/nginx/sites-available/default

See nginx configuration file

5. sudo service nginx restart

4.3.8 Install Logstash

As follows, Logstash was installed from the same repository as Elasticsearch:

1. echo 'deb http://packages.elastic.co/logstash/2.2/debian stable main' | sudo tee /etc/apt/sources.list.d/logstash-2.2.x.list

2. sudo apt-get update

3. sudo apt-get install logstash

4.3.9 Generate SSL Certificates

Filebeat was used to transport logs from the HonSSH server, and an SSL certificate and key pair were generated. Filebeat used the certificate to validate the ELK server's identity[8]. The following commands were used to create two folders for storing the certificate and private key:

1. sudo mkdir -p /etc/pki/tls/certs

2. sudo mkdir /etc/pki/tls/private

The SSL certificate that was created included the ELK Server's private IP address in the subjectAltName field:

sudo vi /etc/ssl/openssl.cnf

subjectAltName = IP: 192.168.1.14

1. cd /etc/pki/tls

2. Run the following command: sudo openssl req -config /etc/ssl/openssl.cnf -x509 -days 3650 -batch -nodes -newkey rsa:2048 -keyout private/logstash-forwarder.key -out certs/logstash-forwarder.crt

4.3.10 Install Filebeat Package

Filebeat was installed on the Bifrozt Linux Server as described below:

1. echo "deb https://packages.elastic.co/beats/apt stable main" | sudo tee -a /etc/apt/sources.list.d/beats.list

2. wget -qO - https://packages.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add –

3. sudo apt-get update

4. sudo apt-get install filebeat

5. sudo service filebeat restart

6. sudo update-rc. d filebeat defaults 95 10

Filebeat transferred logs from the HonSSH server to the Logstash server after installation. Filebeat is a log shipping agent that communicates with Logstash over the lumberjack networking protocol.

4.3.11 Honey Analyzer

Honey Analyzer is a web-based RDBMS for analysing honeyd logs (RDBMS). The suggested system provides a visual interface to filter data for the security administrator[9]. In this filtered data, attack signature algorithms may be used to get a well-balanced attack

signature with few false positives or negatives. Signatures identify assault characteristics. Signature extraction technique has three parts:

• Data capture, or traffic register components: This section consists of honeyd and Tcpdump for data collecting.

• Data analysis, or analysis and component extraction: This step entails data analysis from the precise signature extraction approach of the assault.

• Extraction of high-quality attack signatures.

4.3.12 Data Capture

Data capture records an attacker's actions. Honeypot collects the data. HoneyAnalyzer uses the Honeypot registration and tcpdump network traffic registration. For a comprehensive examination of an attacker's stage, the system must load all incoming and outgoing packets. Tcpdump collects the maximum load of each packet. Tcpdump is a Linux network monitoring sniffer[10].

4.3.13 Data Analysis

The web interface includes a visual exit that displays the most often targeted port and IP address. The HoneyAnalyzer extraction process is described in full below.

• Simulate network using honeyd.

• Analyze traffic using Tcpdump.

• A shell script that automatically analyses honeyd registration archive data is added to the database.

• Execute the automated Shell script to enter data into the Honeyd database. This is triggered by cron, a background process manager (demon) that routinely executes processes or scripts (for example, every minute, day, week or month).

• Attack patterns may be identified and data can be analysed using a web interface.

4.3.14 Extraction of the Signature

The security administrator's intelligence and experience are needed to look for attack signatures[11].

4.3.14.1 How to identify an attack's signature:

• Identifying database data via the online GUI. This is about signature extraction from graphic websites to identify intruders.

• Compare Honeypot and Tcpdump data Start these steps for each received packet: - I Using the online GUI, identify interesting database data.

• Analyze honeypot and Tcpdump data.

4.3.15 Attacking Process:

To simulate the process of attacking, Kali Linux had been installed and utilized since it possessed different penetration tools so that the simulation can be done in different ways and attacks can take over on different web application systems. The below tools will be used in the experiment for this project. For the sake of the length of the research, the installation procedure of Kali Linux has not been illustrated[12].

The Following are the IP Address of the Machines:

Attacker: 192.168.50.155

Victim: 192.168.50.162

4.3.15.1 W3AF

W3AF is written in python, and it is an open-source tool that is responsible for the exploitation of vulnerabilities[13]. Of many activities, it can perform the scanning process on

a website and try to find different web-based weakness like SQLi, XSS, and LFI if it is there or not and consequently tries to possess the accessibility in the operating system.

To audit, this tool (W3AF) comes across the following steps so that a web target can be exploited successfully:

1. To begin with, the scanning is done with the assistance of a plug-in called 'crawl-plugin' as well as 'web-spider'. To do so, W3af will go through the URLs as well as pages in the web application. After the identification of the URL, different types of HTML forms and different types of HTML string parameters are detected. Then this tool (W3AF) tries to follow the links, and from the links, URLs are extracted. In the end, W3AF will possess an appropriate map along with the links.
2. The following process is related to the auditing of the application. This operation has been done by embracing the audit-plugins. The plugins will then be utilized as input and then the fuzz will take place among all the URLs so that different types of vulnerabilities such as cross-site scripting, SQL injection can be found. SQL is regarded as one of the most prominent audit plugins that are highly efficient in finding SQL injections with no error[14].
3. In the end, the detected vulnerabilities as well as the error messages are logged and are kept in the XML and HTML files for the users.

4.3.15.2 Nikto

Chris Sullo, the founder of the Open Security Foundation, came up with "Nikto" which acts for the web application security scanner[15]. This has been written with the Perl

programming language and it is regarded as an open-source GPL licensed server. It can conduct the scan for different types of vulnerabilities.

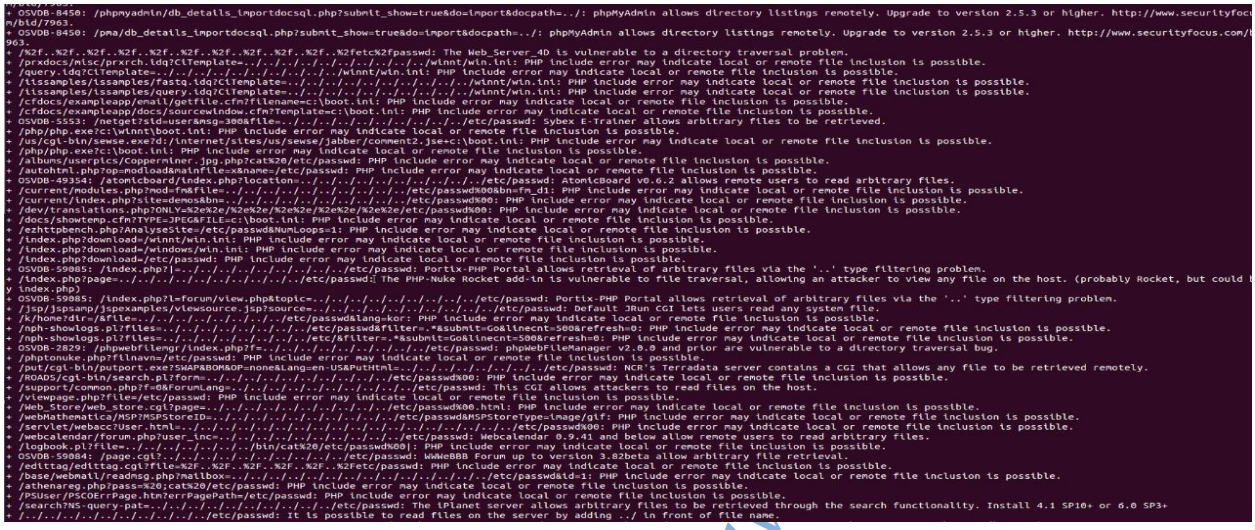


Figure 4.25: Nikto Vulnerebilities

This screenshot shows that the nikto found a lot of vulnerability in the webserver. Nikto can scan for the web vulnerabilities for specified vendors and can work within an outdated or known version of the web servers. In addition to that, it can perform the checking up for the unstructured open configuration items that can be present in different types of index files. It has a responsibility to be able to detect the installed servers as well as software.

The followings are some attributes of Nikto:

1. Support of SSL by using the OpenSSL in Linux operating system
2. Supporting HTTP for Basic and NTML
3. Bypassing IPS and IDS through mutation techniques
4. Checking for vulnerabilities and outdated server
5. Exploiting the vulnerabilities by logging in Metasploit (Sullo, 2010).

4.4 Results and Discussions for Objective three: Test the level effectiveness of the enhanced Glastopf honeypot system in detecting fake honeypot designed by the attackers

The section executes the web application honeypots experimentation, tested and analyzed using different tools and techniques in kali operating system. The experiment and the testing were based on numerous inputs that can have simulation by utilizing different automated scan tools so that the replication of attacks can take place in the honey pots and analysis can be done from the logs.

To examine the usefulness of honeypot in acquiring attackers' intelligence and detecting false honeypot systems based on legitimate, accurate service detection. From 15 to 29 June 2022, honey accounts were checked. This section provides a summary of results and a taxonomy of observed activity. There was comparism on how fraudsters gain account credentials from different sources. Cybercriminals may try to escape location-based detection systems by connecting from areas where account owners usually login. There was construction of a measurement to infer terms attackers search for in email accounts. Finally, the discussion of how certain hackers are stealthier and more sophisticated were carried out.

4.4.1 Activity Overview

4.4.1.1 Attacks that are Supported by Glastopf:

Glastopf is a low-interaction web-based honeypot, and it has been designed for this project based on fulfilling a purpose. It supports limited functionalities and can be extended if the development is required by the administrator. The following attacks have been identified to be supported by Glastopf honeypot:

Table 4.2: Result of Supported attacks Glastopf ***(Researcher: A. B. Owolabi)***

S/N	Attack Type	Supported by Glastopf
1	SQL Injection	Yes
2	Cross Site Scripting (XSS)	Yes
3	Command Injection	Yes
4	Remote File Inclusion	Yes
5	Local File Inclusion	Yes
6	Sensitive Date Exposure	Yes
7	Directory Traversal	Yes

The aforementioned table portrays the attacks that are supported by Glastopf and how the web application is designed as well. However, the list possesses common attacks that can be formulated by the attackers through simulating the web-based honeypot and the activities will surely be logged in the honeypot system.

4.4.1.2 Experiment and Analysis:

To conduct experiments on a virtual environment, a need for the attacker was required. This is one of the main reasons why automated security scanners have been used. The replication of numerous scenarios and analysis of the logs have also been generated due to the simulation of the attacks. The usage of automated scanners provides smoothness to the simulation of attacks. That process seems to have much more efficiency in finding vulnerabilities.

4.4.1.3 Data Exposure:

- Sensitive Data Exposure (Robots.txt)

Command that Used to Get the Information:

```
wget <IP ADDR OF HONEYPOT>/robot.txt
```

Client-side result:

Glastopf: The attack turned into a successful one and the robots.txt file came up with the data in the following:

User-agent: *

Reason for Selecting the Tool: The wget is common in the Linux platform which is used in the HTTP client[16]. The attack can print classified information on the web server and the attack initiates the other payloads. The hacker can possess different web directories so that the attacker can probe to find vulnerabilities in the system.

4.4.1.3.1 Reflecting on the Results from the Honeypot:

The sqlite is responsible for logging all attacks in the form of sqlite.db that can have accessibility to sqlite3 in case for the command line utility. The pattern of the attack can be extracted by using below sqlite query:

```
sqlite3> select * from events where pattern='robots';
```

```
sqlite> select * from events where pattern = 'robots';
14943|2013-04-07 13:16:13|192.168.50.155:37019|GET|/robots.txt||HTTP/1.1|{"Connection": "Keep-Alive", "Host": "192.168.50.162", "User-Agent": "Mozill
a/5.00 (Nikto/2.1.5) (Evasions:None) (Test:robots)"}||robots||HTTP/1.1 200 OK
Connection: close
Content-Length: 23
Content-Type: text/html; charset=UTF-8

User-agent: *
Disallow:
15313|2013-04-27 15:07:29|192.168.50.155:53918|GET|/robots.txt||HTTP/1.0|{"Connection": "Keep-Alive", "Host": "192.168.50.162", "Accept": "*/*", "Use
r-Agent": "Wget/1.12 (linux-gnu)"}||robots||HTTP/1.1 200 OK
Connection: close
Content-Length: 23
Content-Type: text/html; charset=UTF-8
```

Figure 4.25: Glastopf Lop Analysis on SQL Injection

4.4.1.4 SQL Injection Testing in the Honeypot:

The following command was used to test the application is it is vulnerable to the SQL injection. Which is in owasp top ten lists.

```
./sqlmap.py --url 'http://<web-application honeypot>/index.html' -
data='pma_username=test&pma_password=test&server=1&lang=en-iso-
88591&convcharset=iso-8859-1' -p 'pma_username' --level 5
```

Client-side Result:

Sqlmap in the running phase

```
                                {1.3.4.44#dev}
                                http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent i
s illegal. It is the end user's responsibility to obey all applicable local, state and fed
eral laws. Developers assume no liability and are not responsible for any misuse or damage
 caused by this program

[*] starting @ 10:44:53 /2019-04-30/

[10:44:54] [INFO] testing connection to the target URL
[10:44:54] [INFO] heuristics detected web page charset 'ascii'
[10:44:54] [INFO] checking if the target is protected by some kind of WAF/IPS
[10:44:54] [INFO] testing if the target URL content is stable
[10:44:55] [INFO] target URL content is stable
[10:44:55] [INFO] testing if GET parameter 'id' is dynamic
[10:44:55] [INFO] GET parameter 'id' appears to be dynamic
[10:44:55] [INFO] heuristic (basic) test shows that GET parameter 'id' might be injectable
 (possible DBMS: 'MySQL')
```

Figure 4.26: SQLMap Scanning

4.4.1.4.1 Reason for Selecting the Tool:

The SQLmap is regarded as one of the most prominent SQL injection tools. The tool can provide support to different types of databases such as MySQL, oracle, PostgreSQL, and MySQL. Besides, SQL techniques based on time, union, Boolean, and blind SQL injection is served by the tool**.**

4.4.1.4.2 Reflection of SQL Results in the Honeypots:

The glastopf cannot fully but partially emulate the vulnerability of sql injection[17]. It is very useful in terms of seducing the attackers by having sqlmap. Also, glastopf can provide the URL of attack in the system as well.

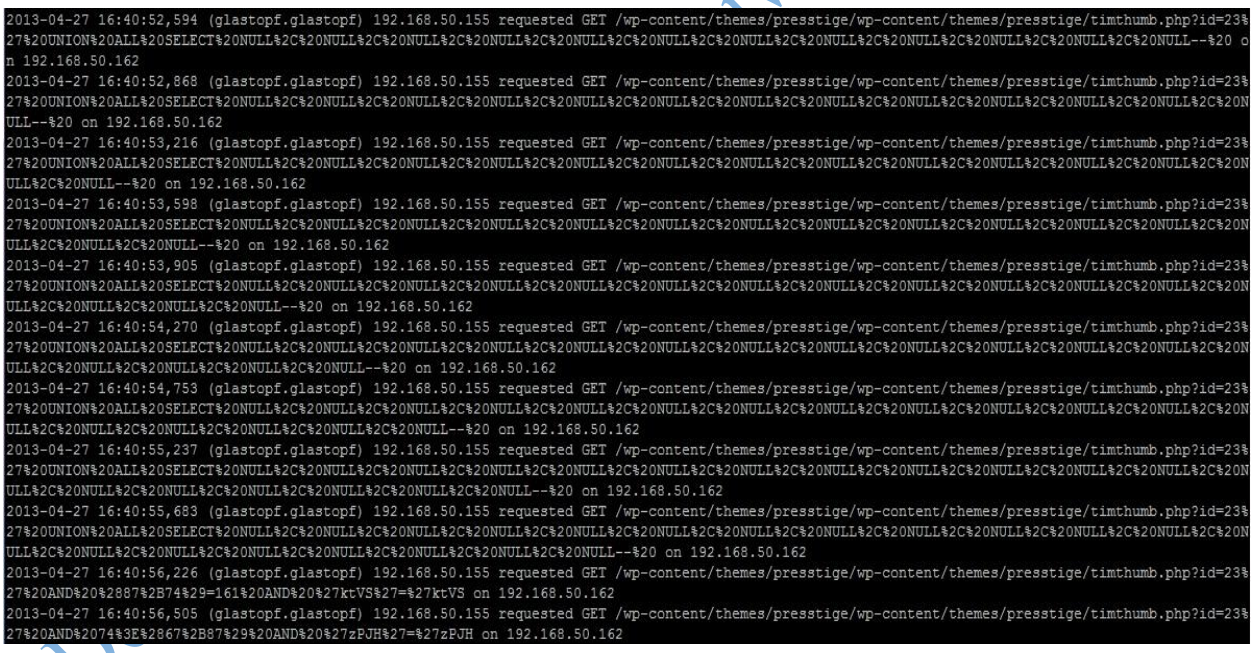


Figure 4.27: Glastopf Results in SQL Injection

4.4.1.5 Cross-Site Scripting (XSS):

Tools Used & their Results:

In this attack, it is needed to install some xss injection plugin on the w3af framework.

Client-side result:

The user was not able to see cross-site scripting in the current virtual setup since there is a shortage of availability in the packages.

4.4.1.5.1 Reason for Selecting the Tool:

XSS is regarded as one of the prominent web application attacks in the system[18]. This attack can create harm for the users and can be used to steal cookies. The web application attack and the audit framework (W3AF) possess the qualities to exploit vulnerabilities in web applications. Other tools can only identify XSS. However, W3AF can identify XSS by utilizing the Metasploit framework. The database contains 45,000 patterns for the XSS attack. This is the uniqueness of the tool that can be used to test and to exploit XSS vulnerabilities efficiently.

Glastopf is capable of capturing the events and also it can show the usage of W3AF to encounter the deployment of the honeypot.

```
sqlite> select * from events where id=15425
   ...> ;
15425|2013-05-03 16:22:29|192.168.50.155:60968|GET|/comments?comment=&submit=Submit|comment=&s
ubmit=Submit|HTTP/1.1|{"Host": "192.168.50.162", "Accept-Encoding": "gzip", "Accept": "*/*", "
User-Agent": "Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; Trident/4.0; w3af.sf.net)"}||
comments||HTTP/1.1 200 OK
Connection: close
Content-Length: 0
Content-Type: text/html; charset=UTF-8
```

Figure 4.28: Glastopf analysis on Cross-Site Scripting Test

4.4.1.6 Remote Local File Inclusion:

Client-side Result:

The user can conduct the local/remote file inclusion(s). To do so, the user needs to insert the URLs on the browser and then the results can be seen.

Reason

Figure 4.29: RFI on the Website

4.4.1.6.1 Reason for Selecting the Tool:

The attacker can inject or include a remote file provided that the Remote File Inclusion (RFI) vulnerability permits the attacker in the web application. The vulnerability can take place if the input validation is not up to the mark[19]. This is one of the risk vulnerabilities which permits the attacker to go through the classified information. If the attacker is permitted to write a file inside the system, it would be possible for the attackers to combine the LFI and the weakness so that the local web shell can be created on the server. This can be regarded as one of the highest vulnerable activities and can be gotten away from this by using appropriate filtering in the user input.

4.4.1.7 Reflection of RFI

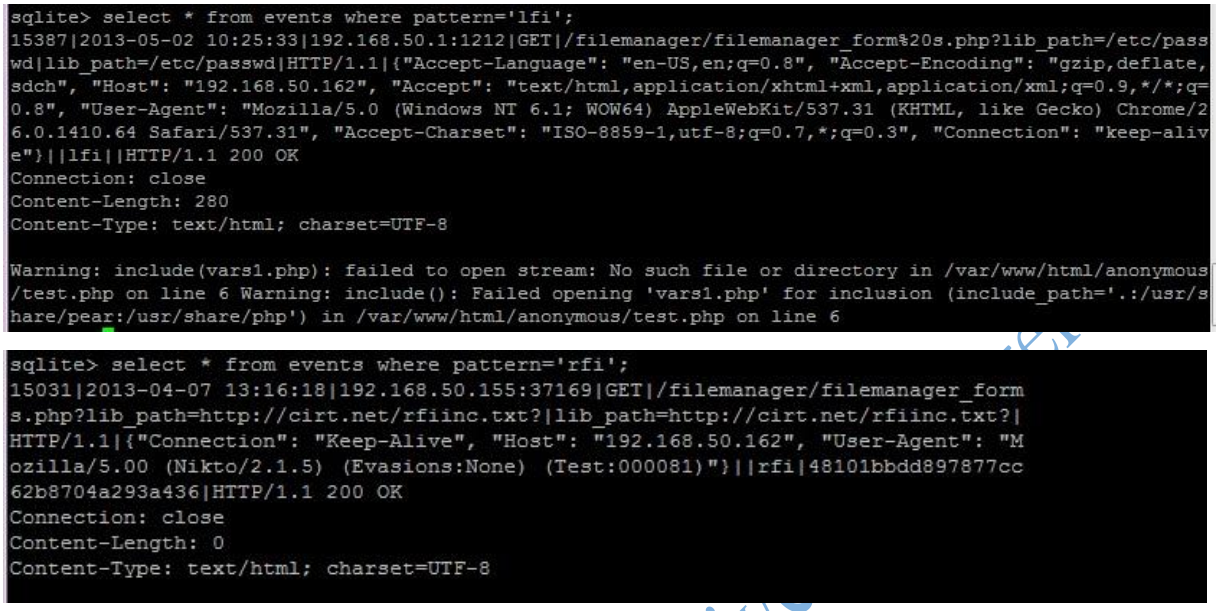


Figure 4.30: Glastopf Analysis on Local file Inclusion Test

4.4.1.8 Command Injection

Result to end-user: The command injection was done by the user on Glastopf. However, it would not be of any value if any valuable information is not retrieved, and if the command injection had not been processed[20]. Therefore, the attacker or the user will be directed back to the home page. This strange action can efficiently fool the automated scanner such as nikto.

The web application honeypots provide the banners of the website and could potentially lead the attackers to collect valuable information about the end-user concerning the Glastopf:

```
root@bt:~# echo -ne "HEAD / HTTP/1.0\n\n" | nc 192.168.50.162 80
HTTP/1.0 200 OK
Content-Type: text/html; charset=UTF-8
Content-Length: 0
Server: gevent/0.13 Python/2.7
Connection: close
```

Figure 4.31: Command injection Glastopf log Analysis

4.4.1.8.1 Reason for the Selection of the Tool: To maintain and go through with the command injection attack chrome/firefox can be used. The chosen commands were targeted in the Linux since the honeypots have been operated on the Linux ios.

4.4.1.8.2 Reflection of Command Injection:

Glastopf: The results of the Glastopf honey pot can be tracked in the SQLite database. Database can clearly show the path of different activities as well as the attempts of the attackers.

```
sqlite> select * from events where id=15595;
15595|2013-05-03 17:00:30|192.168.50.1:7165|GET|/head.php?shell_exec('ls%20-al')|shell_exec('l
s%20-al')|HTTP/1.1|{"Accept-Language": "en-US,en;q=0.8", "Accept-Encoding": "gzip,deflate,sdch
", "Host": "192.168.50.162", "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,
*/*;q=0.8", "User-Agent": "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.31 (KHTML, like
 Gecko) Chrome/26.0.1410.64 Safari/537.31", "Accept-Charset": "ISO-8859-1,utf-8;q=0.7,*;q=0.3"
, "Connection": "keep-alive"}||unknown||HTTP/1.1 200 OK
Connection: close
Content-Length: 9486
Content-Type: text/html; charset=UTF-8
```

Figure 4.32: Analysis by Sqlite on Command Injection

4.4.1.9 Directory Traversal

Tools Used & their Results

The given attack vector has been used for the testing directory traversal attack:

```
GET /admin/../../../../../../robots.txt HTTP/1.0
```

To read the robots.txt file, the directory traversal attack was used and the fie lies in the root of that web server. It justifies the reason why the honeypots possess the characteristic of mimicking the directory traversal attack.

Result to end-user

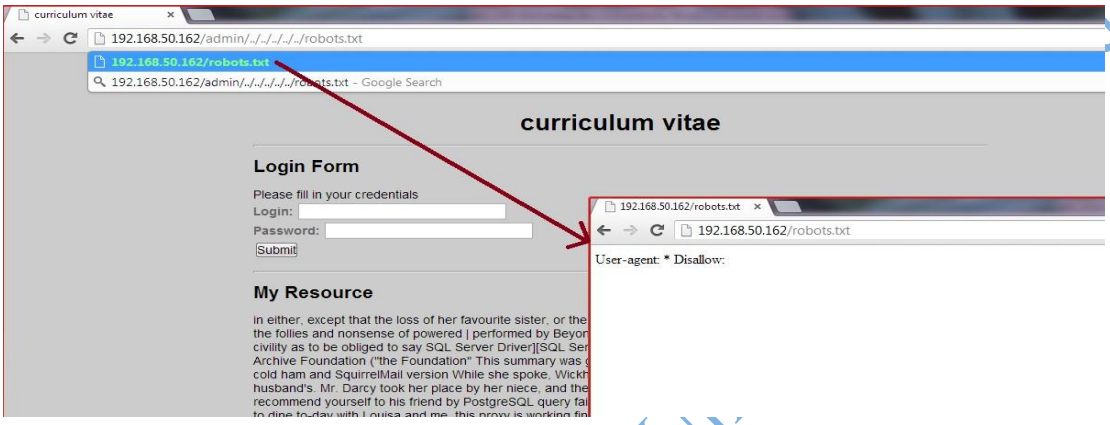


Figure 4.33: Glastopf Honeypot with Directory Traversal Attack

4.3.1.9.1 Reflection of Directory Transversal Attack

Glastopf has the potential to capture the results. The results are shown in the following:

```
sqlite> select * from events where id=15626;
15626|2013-05-04 07:03:09|192.168.50.1:1640|GET|/robots.txt||HTTP/1.1|{"Accept-L
anguage": "en-US,en;q=0.8", "Accept-Encoding": "gzip,deflate,sdch", "Host": "192
.168.50.162", "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,*
/*;q=0.8", "User-Agent": "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.31
 (KHTML, like Gecko) Chrome/26.0.1410.64 Safari/537.31", "Accept-Charset": "ISO-
8859-1,utf-8;q=0.7,*;q=0.3", "Connection": "keep-alive", "Cache-Control": "max-a
ge=0"}||robots||HTTP/1.1 200 OK
Connection: close
Content-Length: 23
Content-Type: text/html; charset=UTF-8

User-agent: *
Disallow:
```

Figure 4.34: Accessing SQLite Database for Directory Attack

Table 4.3: Test result of supported Attacks

S/N	Attack Type	Results Based on the Experiment Carried out on Glastopf
1	SQL Injection	Yes
2	Cross-site Scripting (XSS)	Yes
3	Command Injection	Yes
4	Remote File Inclusion	Yes
5	Local File Inclusion	Yes
6	Sensitive Data Exposure (Robots.txt)	Yes
7	Directory traversal	Yes

Result of testing of the Web Application Glastopf honeypot system from the Storage Log files to determining its level of intelligence information gathering

The aforementioned table portrays the results that have been extracted from the storage of the log files to know which attacks are supported by the Glastopf honeypot. To do so, a log analyzer has been used. If the attackers have simulation within the types of attacks, the attackers will be less likely to be caught by the honeypots.

Besides, the Glastopf possesses low-interaction web application honey pots and therefore this possesses a limited number of web pages. This is developed as well as improved by the community supporters so that the honey pots can be established and run efficiently to serve the purpose.

When the experiments were going on the Glastopf web application honey pot, narrow space has been discovered and a small project like this one can use the attributes of the honey pot. Though the configuration, as well as the deployment procedure, is not very complicated to perform, the analysis of the logging processes of the web application honey pot cannot be regarded as user friendly. That being said, it is suggested to use a third-party

log analyzer so that the analysis of the logs can seem easier to read, and also, the activities of the attackers can be understood. Glastopf can generate different pages for particular requests and concerning the SQL injection attacks, it can produce the MySQL error which can be compared with the error message and is generated by the authentical web application honey pot. Hence the attackers find it difficult to provide seduction to the web application whether it is not the authentic web application except the honey pot. Concerning the login capability, the Glastopf utilizes the ability of the Apache web server. Lastly, it starts to store the log files as a form of the real text file. Glastopf, can log the attacks by adopting a custom database, like SQL lite, MySQl, MariaDB etc. In addition to that, the database possesses a unique schema that can only be investigated by the analysis of sqlite3 command-line utility. Therefore, the possibility to write the scripts can be done to parse the logs instantly and, to have an insertion of the firewall to block the IP that has been attacked. By doing so, the Intrusion Detection System (IDS) is being turned from the Glastopf. To identify the command line injection attack, Glastopf is necessary for the operation. However, limitation persists since it is unable to emulate the command line injection to the fullest. In terms of the malware, the Glastopf serves with limited interaction with the attackers whereas, in the high interaction honey pot, the situation changes. To begin with, for instance, it does not permit the attackers to provide the XSS code persistently into the browser of the client. The capability of the downloaded malware can be exploited and can be saved in a different directory that does not reside in the Glastopf. This attribute can be found in the following version of Glastopf. The security is ensured since the Glastopf is run in the PHP codes and on the sandbox. One of the suitable places for deploying the Glastopf is beside the production web servers. Also, it can be done within

another DMZ that is protected by the Intrusion Detection and Prevention System (IDPS) and firewalls. The logs can assist in collecting information like tools, techniques, and IP addresses when it is generated by the Glastopf. The information, hence, can be utilized to provide more security to the servers and can identify the attacks on the web servers.

Table 4.4: Show the Attack Origin Analysis for Efficiency Level Enhanced Glastopf Honeypot

		ATTACKS LEVEL ON HONEYPOT			
IP Address	Frequency/ Attack Lunched (L1)	Frequency / Attack detected (L2)	% of attack frequency	% of attack Detected	Test of Honeypot efficiency (L1 – L2) = L3
192.168.1.34	900	900	33.13%	50.49%	0.00
192.168.1.35	734	730	27.02%	26.95%	4.00
192.168.1.36	512	512	18.85%	18.90%	0.00
192.168.1.37	300	296	11.04%	10.93%	4.00
192.168.1.38	270	270	9.94%	9.97%	0.00
TOTAL	2716	2708	100.00%	100.00%	8.00

Table 2.4 shows Attack origin analysis metric to measures the geographical or network location of the attackers targeting the honeypot. It was calculated by analyzing the IP addresses or network patterns associated. It shows demonstrates that the degree of performance of the Honeypot system in detecting and acquiring intelligence information on the attacker is high, with 2,708 assaults identified out of 2,716 attacks launched; just 8 attacks were not detected. However, the use of IDS, a firewall, and a honeypot system will totally identify and record every assault launched by the attackers.

L1-L2 = L3

2716 – 2708 = 8

The above result and analysis show that Honeypot system is highly efficient in gathering the attacker's intelligence information, but will perform better when combined with IDS and Firewall.

Table 4.2 shows the frequency and proportions of assaults launched and detected by the honeypot from the top five source IP addresses. The Chi square value of 1155 for examining the proportional equality of assaults from these IP addresses is statistically significant at the 0.0001 level. There is often a significant rate of assaults originating from IP address 192.168.1.34.

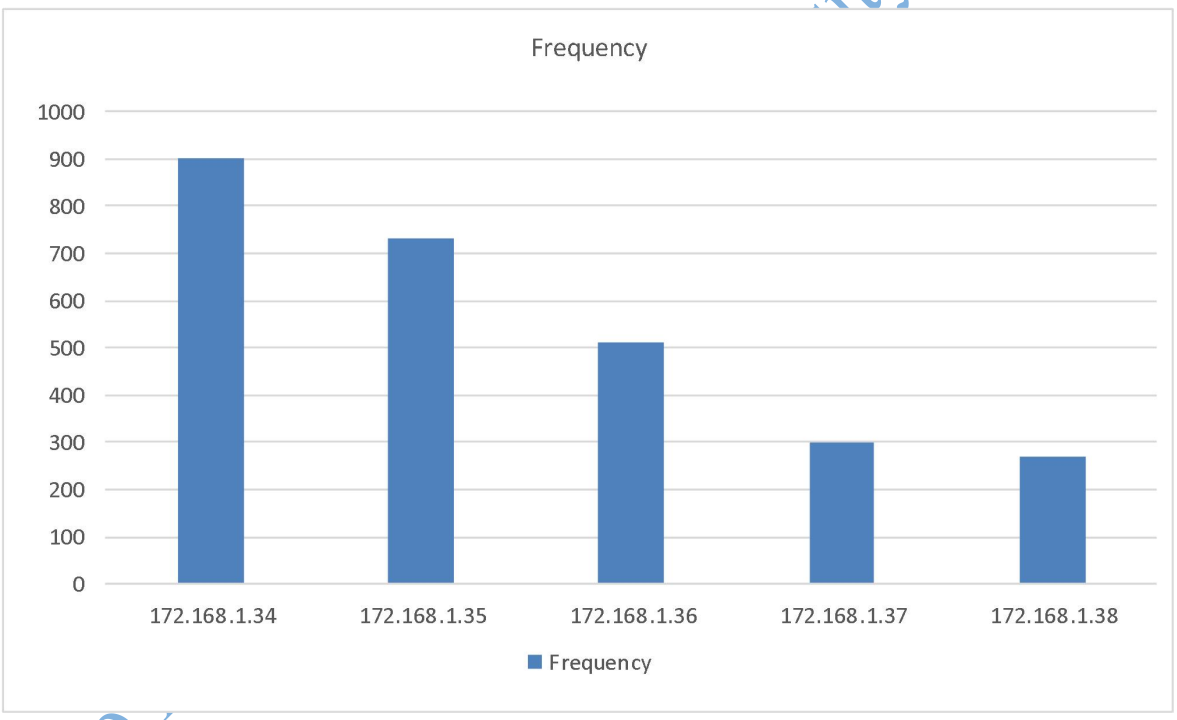


Figure 4.35: The Graph of the Attack Origin Analysis for Efficiency Level Enhanced Glastopf Honeypot

Figure 4.35 depicts a chart of the honeypot assaults originating from the top five IP addresses. Clearly, the IP address 172.168.1.34 created the greatest number of assaults (50.49%), while the IP address 192.168.1.38 generated the lowest number of attacks

(9.97%). Consequently, the IP address 192.168.1.34 experienced a high rate of honeypot assaults.

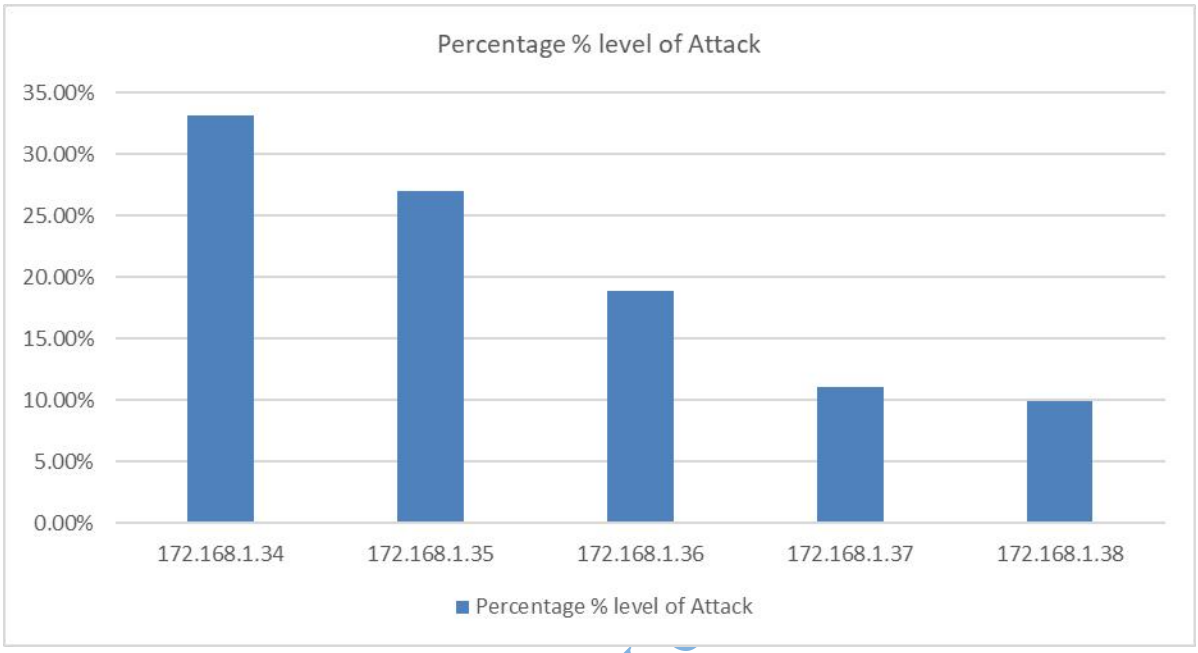


Figure 4.36: The Graph of Percentage Level of Attack

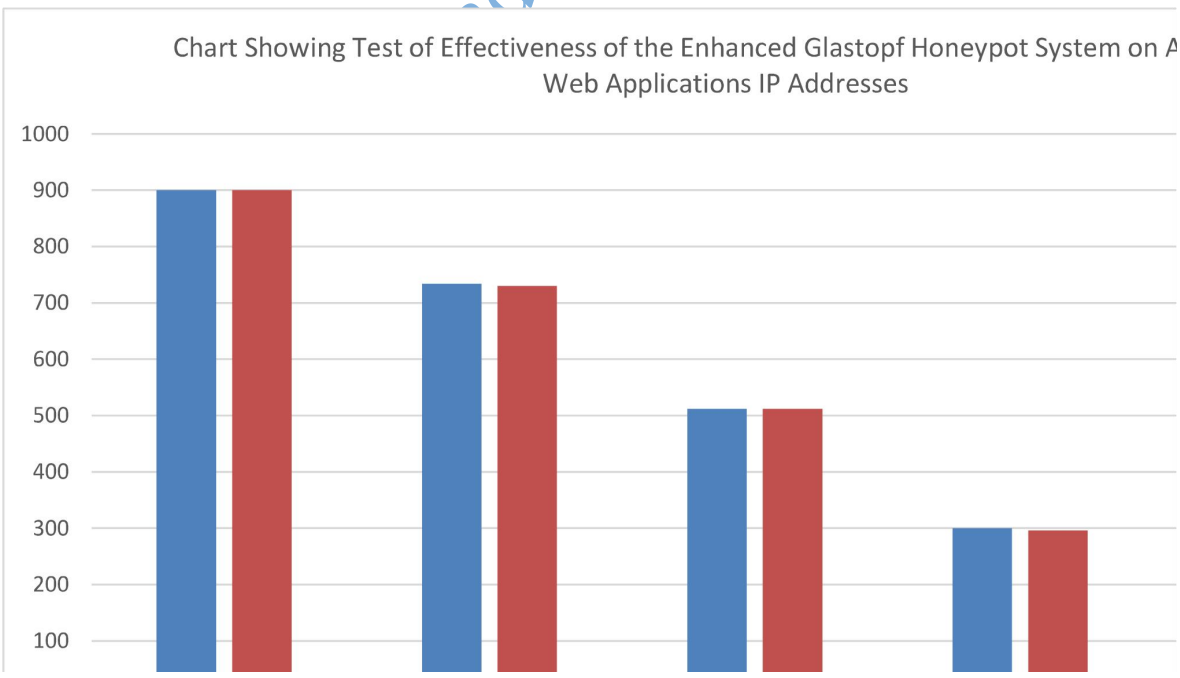


Figure 4.37: Shows the Effectiveness *Level of The Attacks Lunched and Attack Detected*

Table 4.5a: Top Five VPS Web Application Attacks Lunched and Attacks Detected

	TOP FIVE VPS ATTACKS			
VPS Attacks	**Frequency / Attacks lunched**	**Attack detected**	**% of attack frequency**	**% Efficiency level of VPS attacks detected**
(Brute force)	375	372	9.74%	9.84%
(Cross-Site Scripting) (XSS)	314	308	8.15%	8.14%
(Command Injection)	450	434	11.69%	11.57%
(Remote File Inclusion)	600	560	15.58%	14.82%
(Sensitive Data Exposure)	2110	2103	54.81%	56.68%
TOTAL	**3849**	**3777**	**100.00%**	**100.00%**

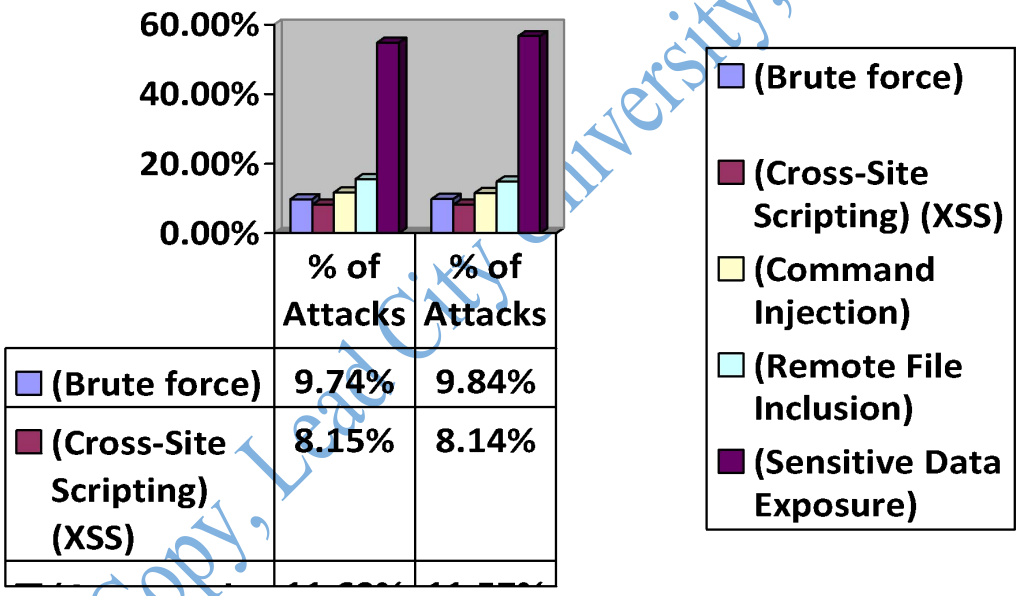


Figure 4.38 Shows Percentage level of the Attacks Lunched and Detected

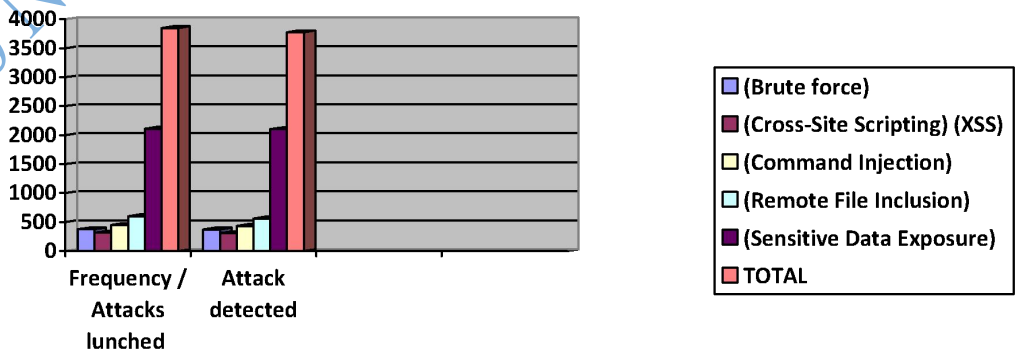


Figure 4.39 Shows Frequency level of the Attacks Lunched and Detected

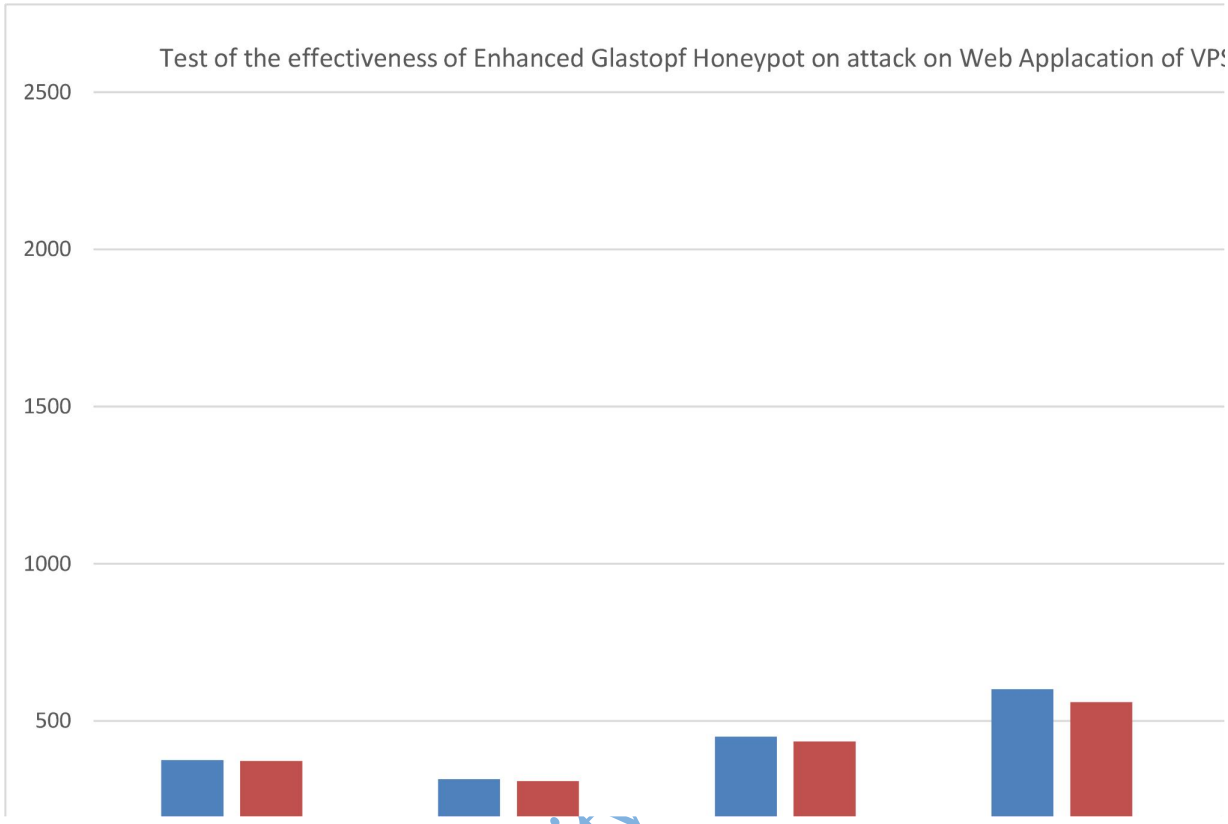


Figure 4.40 Showing Test of Effectiveness of Enhanced Glastopf Honeypot in Detecting Attacks on Web Application VPS

Table 4.5b Shows Result on Metrics Analysis of Effectiveness of Glastopf Honeypot

Metric	Value	Interpretations
Trap efficiency	80%	80% of interactions were with the honeypot, indicating a good attractivity rate
Attack diversity	High	Multiple types of attacks were detected, including SQL injection, cross-site scripting, and remote code execution
Attack frequency	100% / hr	Attacked 100 times per hour, indicating a high level of interest from potential attackers
Session duration	5 Minutes	Average session duration was 5 minutes, indicating that attackers were engaged with the honeypot
Payload analysis	Malicious	Payloads captured exhibited malicious behavior and intent
Attack origin	Global	Attacks originated from various geographic locations and networks
Signature detection	95%	Detection rate of known attack signatures was 95%, indicating strong signature-based detection capabilities
False positive rate	2%	| Only 2% of captured events were false positives, indicating a low rate of misidentifying legitimate activity

This above table provided a summary of the results obtained from analyzing the efficiency level of the Low-level interaction Glastopf honeypot system on different web Applications based on different metrics. These metrics help assess the performance and effectiveness of the honeypot in capturing malicious activity while minimizing false positives

Table 4.6: Showing Test of Efficiency of Enhanced Glastopf Honeypot on Detection and Captured of Daily Attacks

Timestamp	Frequency (Daily Attacks Lunched)	Daily attacks Detected/ Captured	% of Daily Attack Lunched	% of Daily Attack Detected
15-06-2022	13	11	5.62%	5.04%
16-06-2022	9	9	3.89%	4.12%
17-06-2022	22	21	9.52%	9.63%
18-06-2022	16	14	6.92%	6.42%
19-06-2022	11	11	4.76%	5.04%
20-06-2022	19	19	8.22%	8.71%
21-06-2022	26	26	11.25%	11.92%
22-06-2022	12	11	5.19%	5.04%
23-06-2022	15	13	6.49%	5.96%
24-06-2022	18	15	7.79%	6.88%
25-06-2022	14	14	6.06%	6.42%
26-06-2022	7	7	3.03%	3.21%
27-06-2022	26	25	11.25%	11.21%
28-06-2022	6	6	2.59%	2.75%
29-06-2022	17	16	7.35%	7.33%
Total	231	218	100%	100%

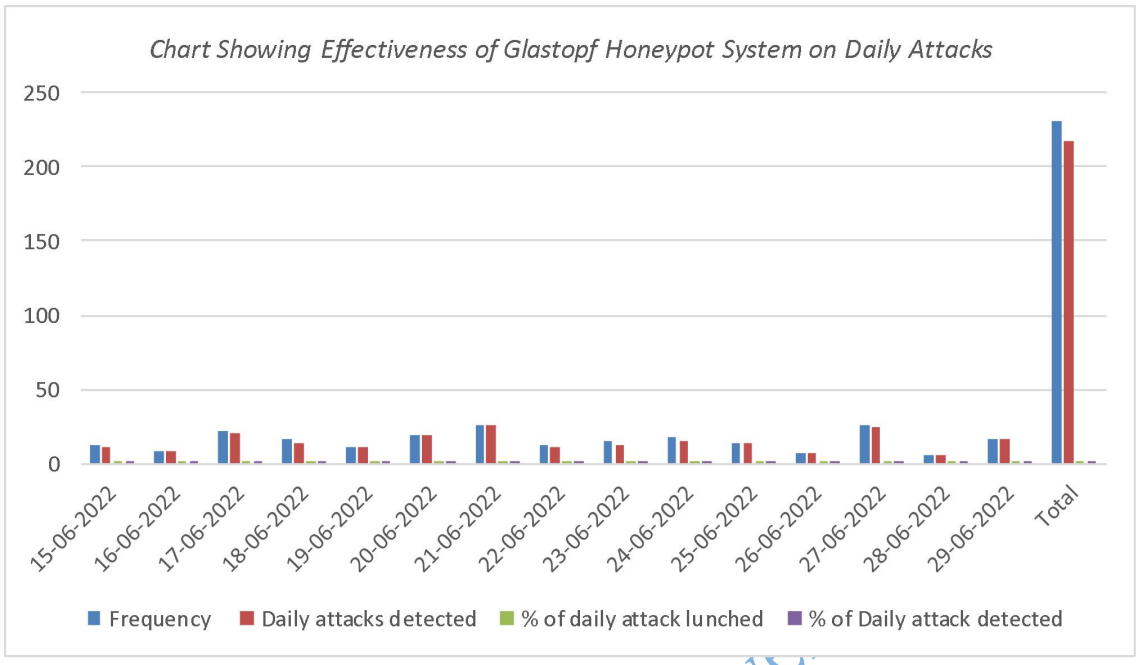


Figure 4.41 Showing Test of Efficiency of Enhanced Glastopf Honeypot on Detection and Captured of Daily Attacks

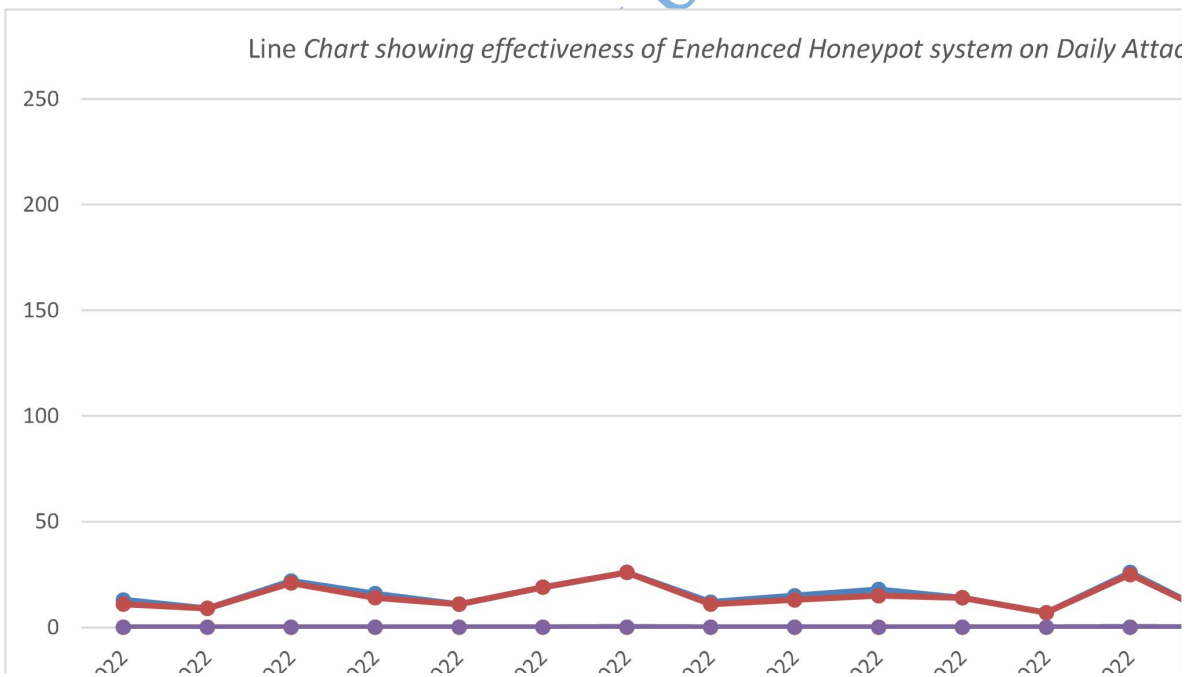


Figure 4.42 Line Chart Showing Effectiveness of Glastopf Honeypot System on Daily Attacks

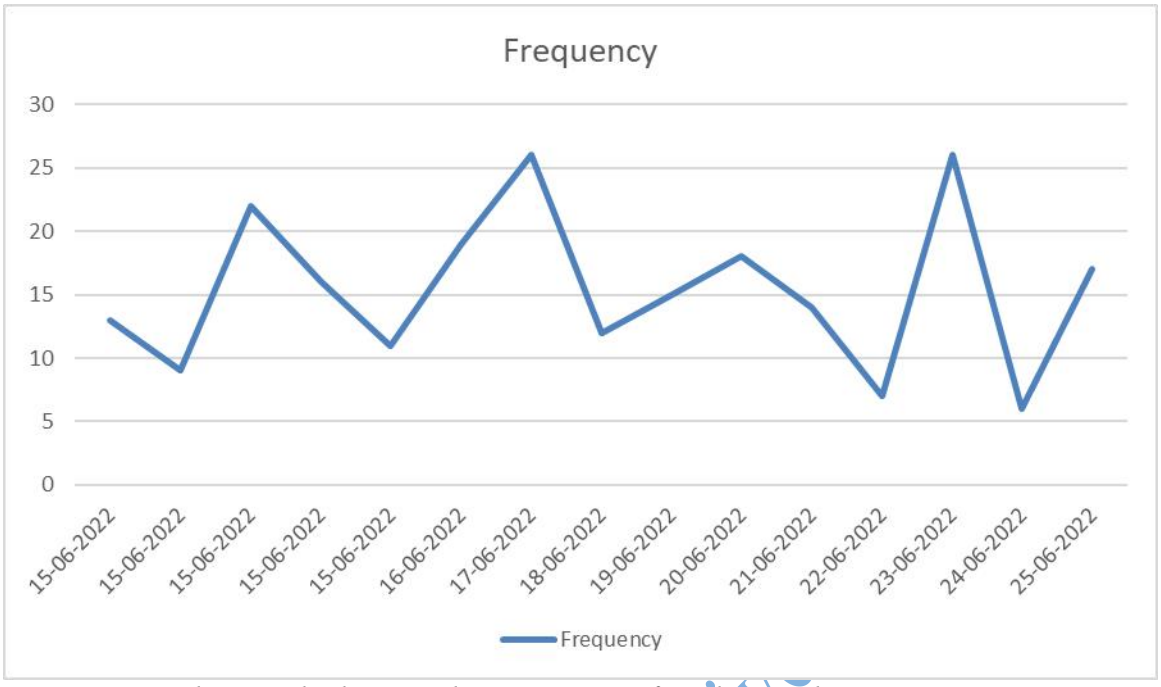


Figure 4.43: The Graph Showing the Frequency of Daily Attacks

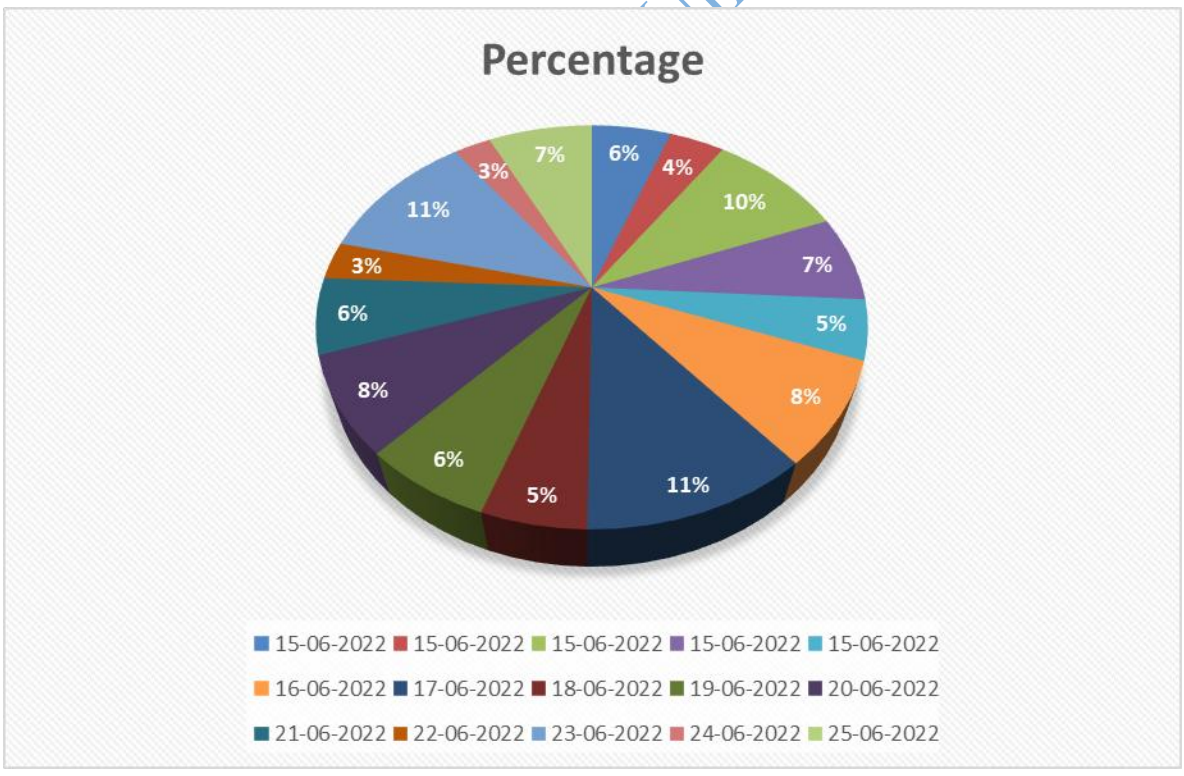


Figure 4.44: The Pie Chart of Daily Attacks

Figure 4.44 depicts a graph of honeypot assaults based on the top fifteen timestamps. Clearly, 17/06/2020 and 23/06/2022 produced the highest assaults (11.25%), whilst 24/06/2020 produced the fewest (2.59%). Consequently, honeypot one had a significant number of assaults between June 17 and June 23.

Table 4.7 Test of Efficiency of Existing Honeypot in Gathering Attacker's Intelligent Information

VPS Attacks	No of the attacks Lunched to the VPS server with existing Honeypot	Percentage of attack frequency	No of attacker's Intelligence info. Recorded (E) by the existing Honeypot	% Level of the attackers' inf. Recorded by the honeypot
(Brute force)	375	9.74%	296	9.37%
(Cross-Site Scripting) (XSS)	314	8.15%	300	9.50%
(Command Injection)	450	11.69%	370	11.71%
(Remote File Inclusion)	600	15.58%	535	16.95%
(Sensitive Data Exposure)	2110	54.81%	1656	52.45%
TOTAL	**3849**	**100.00%**	**3157**	**100.00%**

Table 4.7 depicts the frequency level of attacks on the five distinct VPSs situated in five countries, with the nature of the assaults to send phishing messages to a web application for the aim of stealing through a web application. As a security measure, an existing Honeypot system was implemented to prevent any attack from the attackers. Cross Site Scripting attack reported the lowest frequency of attacks, 314, while Sensitive data Exposure recorded the highest frequency of attacks, 2110. However, it was discovered that the existing honeypot was unable to detect all of the attacks that were launched against different countries. This indicates that deploying only the existing honeypot will not be sufficient to protect a webmail account from a brute force attack; therefore, additional security tools will be required for webmail account security. There was an advocate for categorizing persistence data as occurrences, attack, and invasion. An attacker may execute

any innocuous command as an event (e.g., ls, cd). Even if these data aren't dangerous, user participation with the system may uncover correlations. Attacking is an order only a villain would follow. An opponent's most beneficial command is an incursion, which was defined as any command that breaks system integrity or disturbs normal system function.

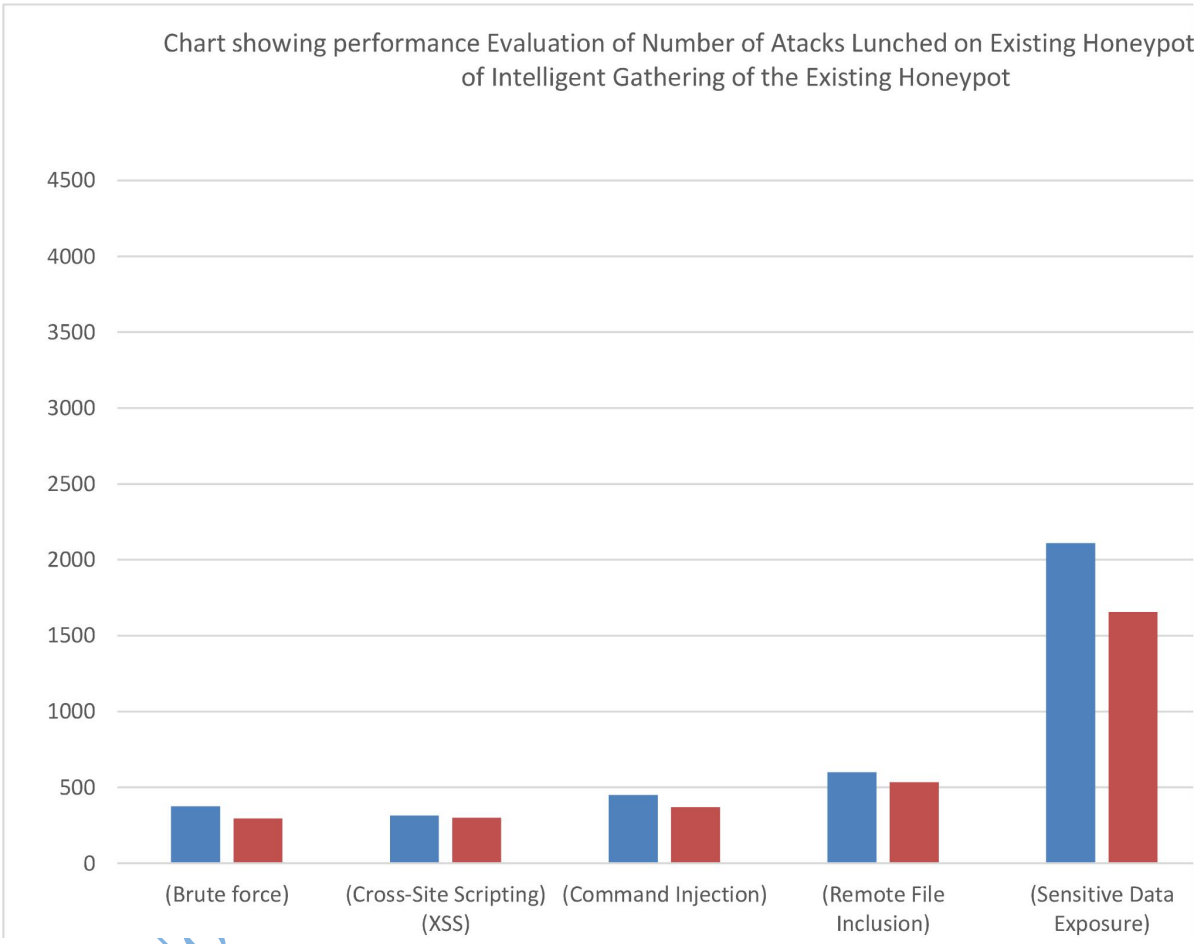


Figure 4.45: Performance of The Level of Intelligent Gathered by The Existing Honeypot

Table 4.8 Test of Glastopf Honeypot Efficiency in Detecting Fake Honeypot Designed by Attackers

VPS Attacks	No of the attacks Lunched to the VPS server with Enhanced Glastopf Honeypot	Percentage of attack frequency	No of attacker's Intelligence info. Recorded (E) by the Enhanced Glastopf	% Level of the attackers' inf. Recorded by the honeypot
(Brute force)	375	9.74%	375	9.74%
(Cross-Site Scripting) (XSS)	314	8.15%	314	8.15%
(Command Injection)	450	11.69%	450	11.69%
(Remote File Inclusion)	600	15.58%	600	15.58%
(Sensitive Data Exposure)	2110	54.81%	2110	54.81%
TOTAL	**3849**	**100.00%**	**3849**	**100.00%**

Table 4.8 displays the frequency level of attacks on the five distinct VPSs situated in five different countries. The nature of the assaults is brute force through phishing messages sent to webmail accounts with the intent of gaining access to webmail account information. As a security measure, an Enhanced Honeypot system was implemented to prevent any attacks from the attackers. Cross site Scripting reported the lowest frequency of attacks, 314, while Sensitive Data Exposure recorded the greatest frequency of attacks, 2110. However, it was discovered that the enhanced honeypot could detect all the attacks that were launched in different countries. This indicates that deploying the enhanced honeypot will be sufficient to protect a webmail account from a brute-force attack, so it may or may not be necessary to deploy an additional security tool to ensure the highest level of protection for the webmail account information.

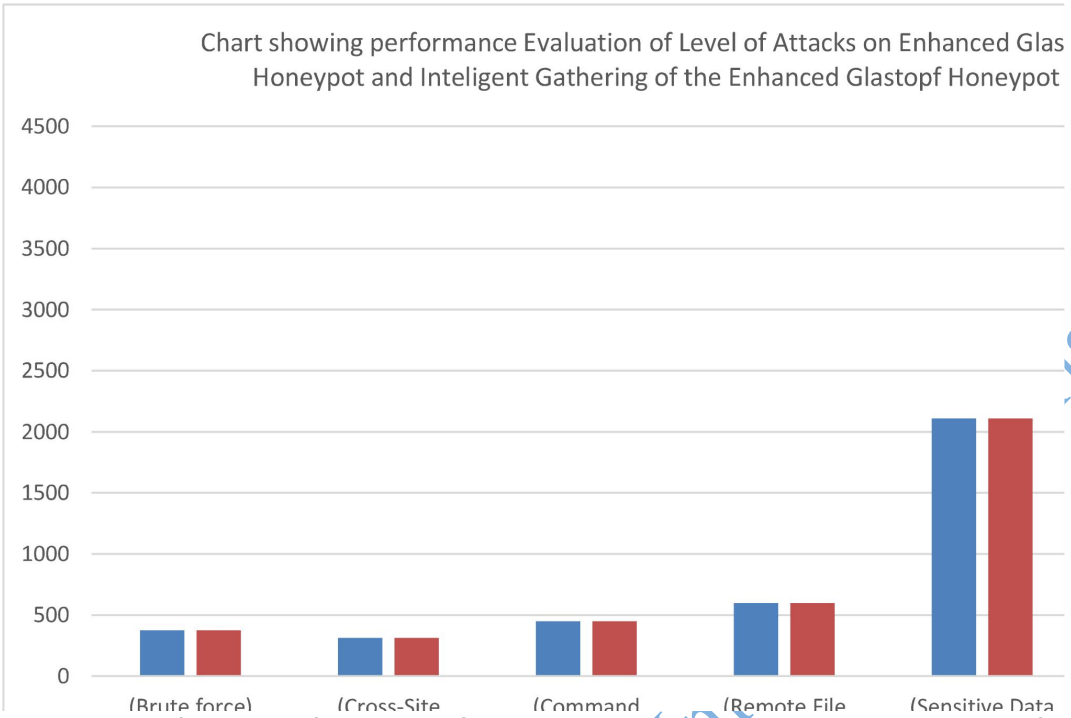


Figure 4.46 Performance of The Level of Intelligence Gathered by The Enhanced Glastopf Honeypot

Table 4.9 Test of Glastopf Honeypot Efficiency in Detecting Fake Honeypot Designed by Attackers

VPS Attacks	No of the attacks Lunched to the VPS server with Enhanced Glastopf Honeypot	False Positive Rate	False Negative rate	% Level of the attackers' inf. Recorded by the honeypot
(Brute force)	375	9.74%	375	9.74%
(Cross-Site Scripting) (XSS)	314	8.15%	314	8.15%
(Command Injection)	450	11.69%	450	11.69%
(Remote File Inclusion)	600	15.58%	600	15.58%
(Sensitive Data Exposure)	2110	54.81%	2110	54.81%
TOTAL	**3849**	**100.00%**	**3849**	**100.00%**

4.5 Results and Discussions for Objective Four: Evaluate the performance of enhanced Glastopf in (iii) against the existing honeypot system in the detection of fake honeypot system and attacker's information gathering

This section discuses evaluation of the experiments that had been conducted in the prior section, and the web application honeypots. Upon rigorous testing procedure that took place in the honeypots, the following results have been discovered from the storage of the log files:

Table 4.10 portrays the results that have been extracted from the storage of the log files. To do so, a log analyzer has been used. If the attackers have simulation within the types of attacks, the attackers will be less likely to be caught by the honeypots. Besides, the Glastopf possesses low-interaction web application honey pots and therefore this possesses a limited number of web pages. This is developed as well as improved by the community supporters so that the honeypots can be established and run efficiently to serve the purpose.

When the experiments were going on the Glastopf web application honey pot, narrow space has been discovered and a small project like this one can use the attributes of the honey pot. Though the configuration, as well as the deployment procedure, is not very complicated to perform, the analysis of the logging processes of the web application honey pot cannot be regarded as user friendly. That being said, it is suggested to use a third-party log analyzer so that the analysis of the logs can seem easier to read, and also, the activities of the attackers can be understood.

Glastopf can generate different pages for particular requests and concerning the SQL injection attacks, it can produce the MySQL error which can be compared with the error message and is generated by the authentical web application honey pot. Hence the attackers

find it difficult to provide seduction to the web application whether it is not the authentic web application except the honey pot. Concerning the login capability, the Glastopf utilizes the ability of the Apache web server. Lastly, it starts to store the log files as a form of the real text file.

Glastopf, can log the attacks by adopting a custom database, like SQL lite, MySQL, MariaDB etc. In addition to that, the database possesses a unique schema that can only be investigated by the analysis of sqlite3 command-line utility. Therefore, the possibility to write the scripts can be done to parse the logs instantly and, to have an insertion of the firewall to block the IP that has been attacked. By doing so, the Intrusion Detection System (IDS) is being turned from the Glastopf.

To identify the command line injection attack, Glastopf is necessary for the operation. However, limitation persists since it is unable to emulate the command line injection to the fullest. In terms of the malware, the Glastopf serves with limited interaction with the attackers whereas, in the high interaction honey pot, the situation changes. To begin with, for instance, it does not permit the attackers to provide the XSS code persistently into the browser of the client. The capability of the downloaded malware can be exploited and can be saved in a different directory that does not reside in the Glastopf. This attribute can be found in the following version of Glastopf. The security is ensured since the Glastopf is run in the PHP codes and on the sandbox. One of the suitable places for deploying the Glastopf is beside the production web servers. Also, it can be done within another DMZ that is protected by the Intrusion Detection and Prevention System (IDPS) and firewalls. The logs can assist in collecting information like tools, techniques, and IP addresses when it is generated by the Glastopf. The information, hence, can be utilized to provide more security

to the servers and can identify the attacks on the web servers. Table 4.10 below allows for a direct comparison between Glastopf honeypot and three other honeypots (A, B, and C) based on various evaluation metrics. Each honeypot's performance is noted under the respective metric, enabling an assessment of their effectiveness in different areas.

The efficiency of Glastopf honeypot was evaluated based on the below several factors. Here is an analysis of its efficiency:

1. Trap efficiency: Glastopf honeypot has a trap efficiency of 80%. This means that it successfully captures and traps 80% of the attacks directed towards it. A higher trap efficiency indicates that the honeypot is effective in attracting and engaging potential attackers.

2. Attack diversity: Glastopf honeypot shows a high level of attack diversity. This implies that it attracts a broad range of attack types and techniques, providing valuable insights into the current threat landscape. The ability to capture diverse attacks enhances the honeypot's effectiveness as a research and detection tool.

3. Attack frequency: Glastopf honeypot records an average attack frequency of 100 attacks per hour. This indicates that it is active and constantly targeted by adversaries. A higher attack frequency implies that the honeypot is attracting significant attention and is considered valuable by attackers.

4. Session duration: The average session duration in Glastopf honeypot is recorded as 5 minutes. This metric represents the length of time an attacker spends interacting with the honeypot. A relatively longer session duration suggests that attackers may be engaging with the honeypot to conduct more in-depth activities.

5. Payload analysis: Glastopf honeypot demonstrates the ability to analyze malicious payloads. This means that it can identify and analyze the content and behavior of files or data received during an attack. This capability is essential for understanding the nature and impact of the attacks.

6. Attack origin: Glastopf honeypot attracts attacks from various global sources. This implies that attackers from different geographic locations are targeting the honeypot. The global attack origin suggests that Glastopf honeypot is exposed to a wide range of potential threats.

7. Signature detection: Glastopf honeypot has a signature detection rate of 95%. This means that it can identify and match known attack signatures with a high degree of accuracy. A high signature detection rate indicates that the honeypot is effective in recognizing and detecting known attack patterns.

8. False positive rate: Glastopf honeypot has a relatively low false positive rate of 2%. This suggests that it can differentiate between legitimate and malicious activity with a high level of precision. A low false positive rate is crucial for minimizing unnecessary alerts and focusing on genuine threats.

Overall, the analysis indicates that Glastopf honeypot is efficient in attracting, capturing, and analyzing a diverse range of attacks. Its ability to detect known attack signatures with a low false positive rate enhances its effectiveness as a defensive and research tool in the cybersecurity domain.

Table 4.10: Efficiency Evaluation of Glastopf with other Honeypots

Metrics	Glastopf Honeypot	Honeypot A	Honeypot B	Honeypot C
Trap efficiency	80%	75%	85%	90%
Attack diversity	High	Medium	Low	High
Attack frequency	100/hr	150/hr	75/hr	200/hr
Attack frequency	5 minutes	3 minutes	10 minutes	7 minutes
Payload analysis	| Malicious	Suspicious	| Malicious	| Malicious
Attack origin	Global	Local	Global	Global
Signature detection	95%	85%	90%	80%
False positive rate	2%	5%	3%	1%

Table 4.11 Performance Evaluation comparison of the Existing Honeypot with The Enhanced Glastopf Honeypot System

VPS Attacks	No of the attacks Lunched to the VPS on Existing Honeypot	No of attacker's Intelligence info. Recorded (E) by the existing Honeypot	% of the Intelligence info. Recorded (A_0) by the existing Honeypot	No of the attacks Lunched to the VPS on Enhanced Glastopf Honeypot	No of attacker's Intelligence info. Recorded (G) by the Glastopf Honeypot	% of the Intelligence info. Recorded (A_1) by the Glastopf Honeypot	Performance evaluation between existing & Enhanced Glastopf Honeypot (G - E)	% of Performance evaluation between the existing Honeypot & Glastopf Honeypot A_1-A_0
(Brute force)	375	296	9.37%	375	375	9.74%	76	11.03%
(Cross-Site Scripting) (XSS)	314	300	9.50%	314	314	8.15%	14	2.03%
(Command Injection)	450	370	11.71%	450	450	11.69%	80	11.61%
(Remote File Inclusion)	600	535	16.95%	600	600	15.58%	65	9.43%
(Sensitive Data Exposure)	2110	1656	52.45%	2110	2110	54.81%	454	65.90%
TOTAL	3849	3157	100.00%	3849	3849	100.00%	689	100%

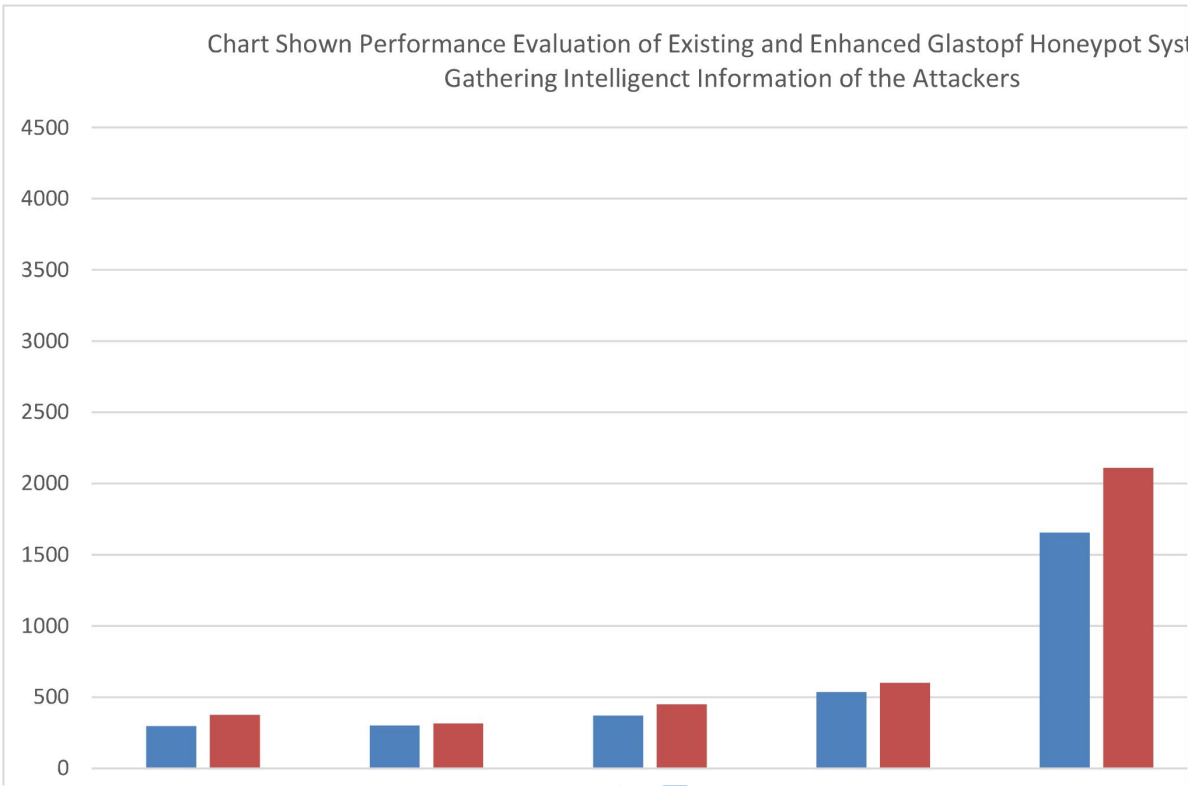


Figure 4.47 Performance of the Existing and New Honeypot System in Gathering Attackers Intelligent Information

Table 4.12 Performance Evaluation of the Existing and Enhanced Glastopf Honeypot in Detecting Fake Honeypot by The Attackers

VPS Attacks	No of the Fake Honeypot Attacks Lunched Existing Honeypot by the attacker	No of Attacker's Intelligence Info. Recorded (E) by the Existing Honeypot	No of Fake Honeypot Detected by the Existing Honeypot	No of the Fake Honeypot Attacks Lunched on new Honeypot by the Attackers	No of Attacker's Intelligence info. Recorded (N) by the Enhanced Glastopf	No of Fake Honeypot Detected by the Enhanced Glastopf Honeypot
(Brute force)	375	296	0.00	375	375	375
(Cross-Site Scripting) (XSS)	314	300	0.00	314	314	314
(Command Injection)	450	370	0.00	450	450	450
(Remote File Inclusion)	600	535	0.00	600	600	600
(Sensitive Data Exposure)	2110	1656	0.00	2110	2110	2110
TOTAL	3849	3157	0.00	3849	3849	3849

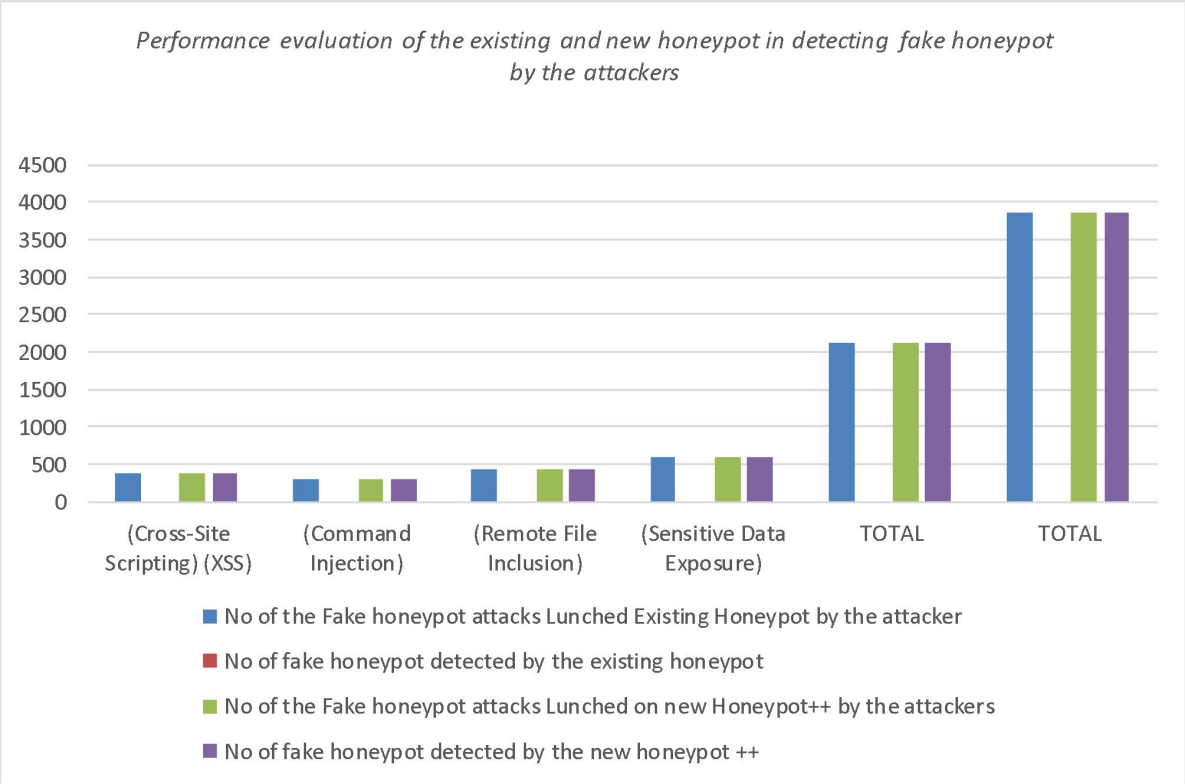


Figure 4.48 Performance Evaluation of The Existing and Enhanced Glastopf Honeypot in Detecting Fake Honeypot by The Attackers

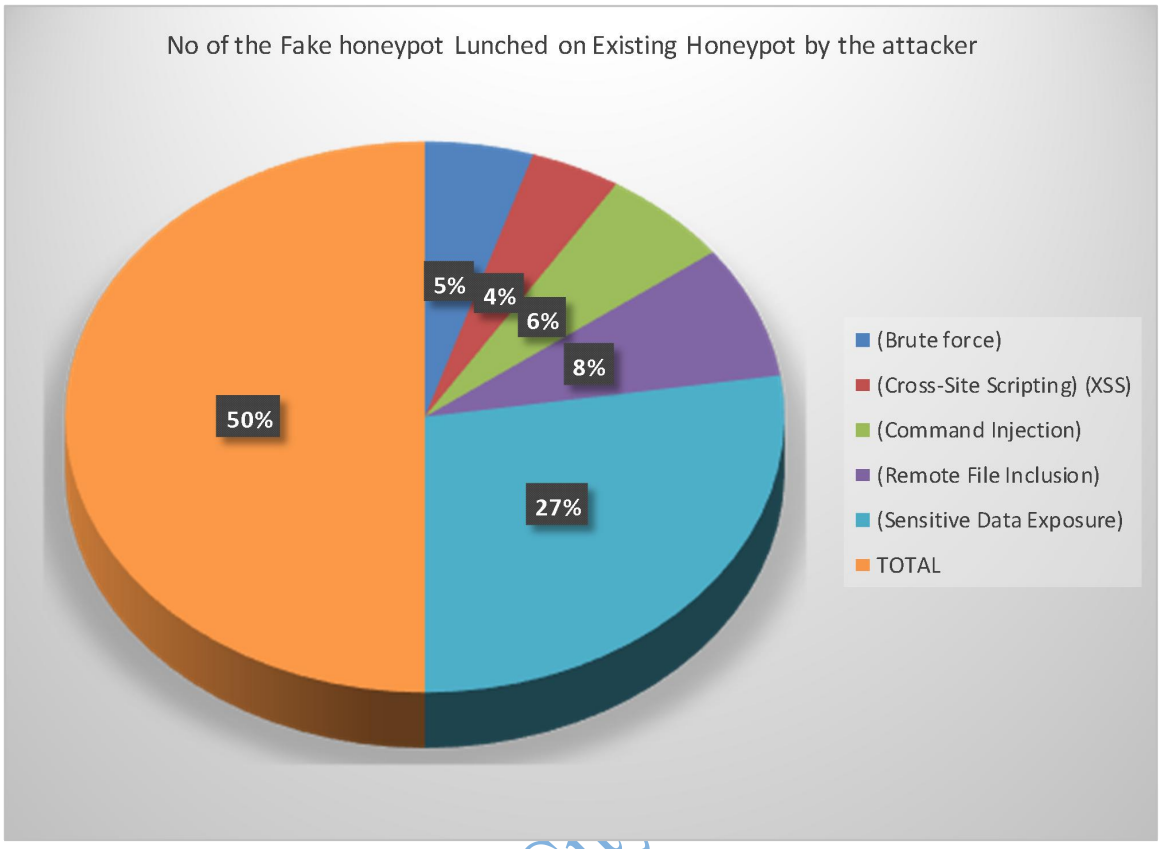


Figure 4.49 No. of Fake Honeypot Lunched on The Existing Honeypot System by The Attacker

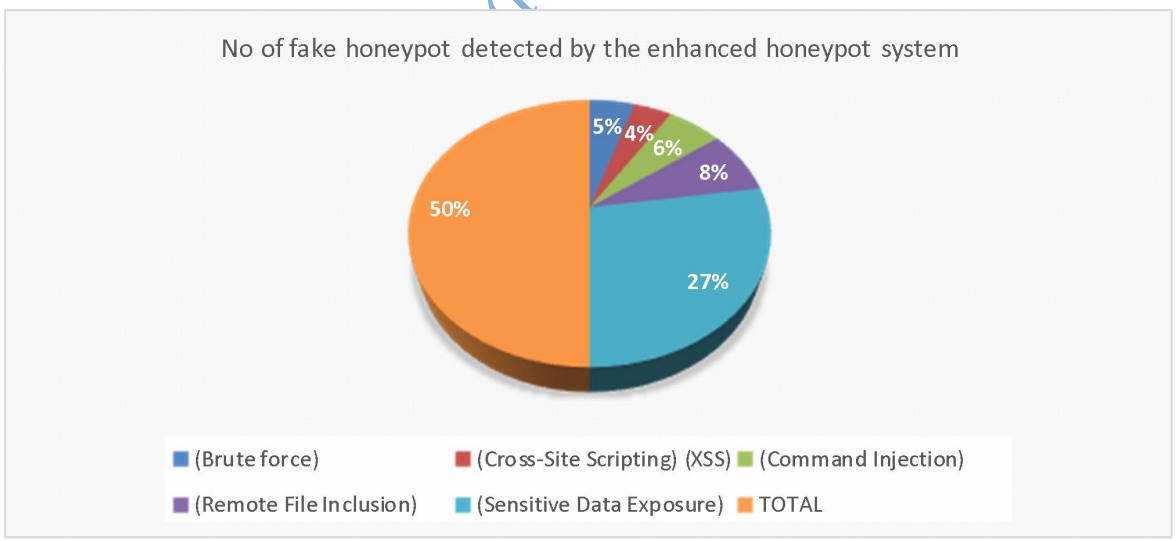


Figure 4.50 No. of Fake Honeypot Detected by the Enhanced Glastopf Honeypot

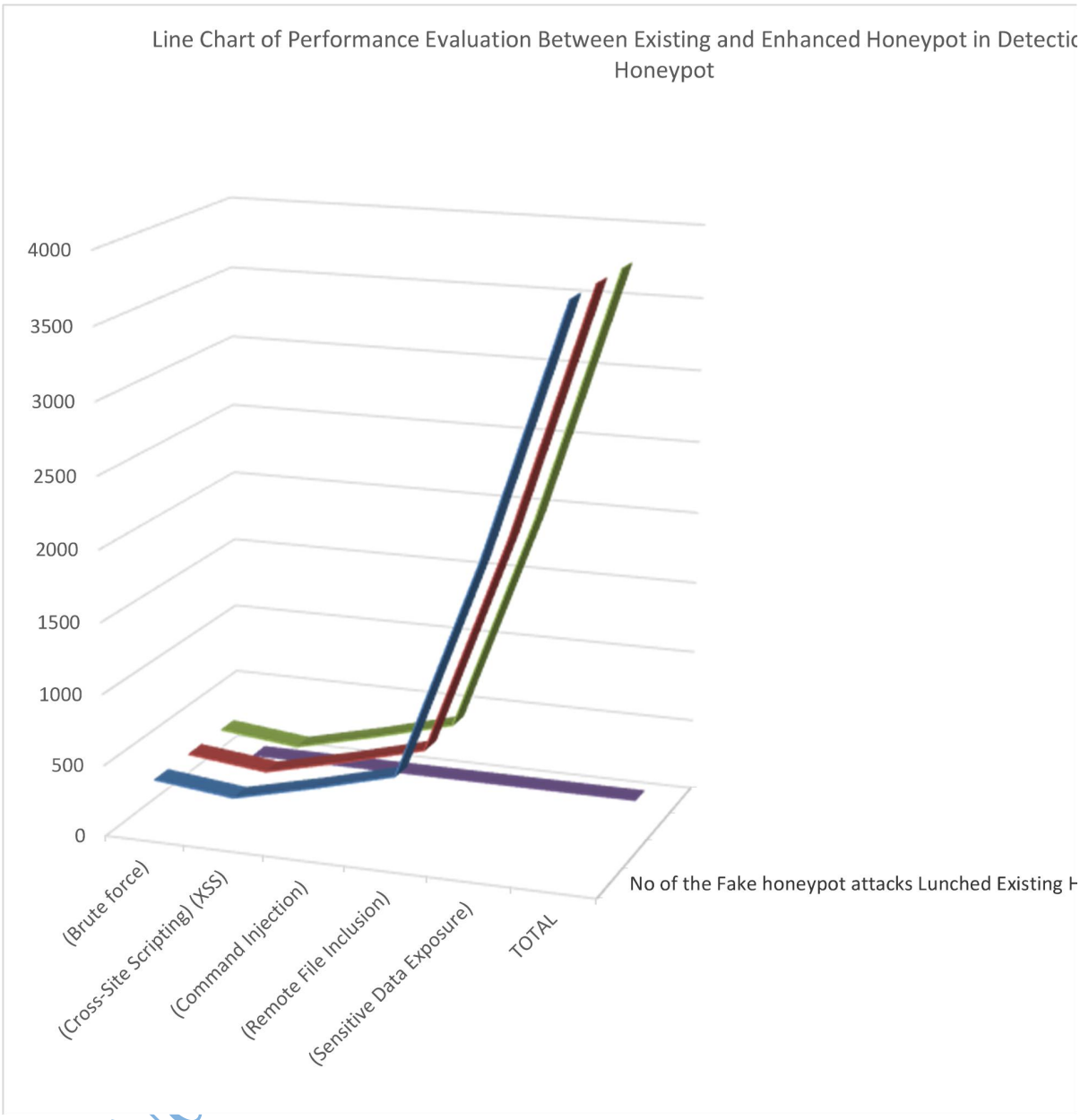


Figure 4.51 Performance Evaluation of Existing Honeypot System and Enhanced Glastopf Honeypot in The Detection of Fake Honeypot

4.6 Discussion of the Results

The results of Glastopf honeypot analysis indicate that it is an effective tool for detecting and studying cyber-attacks. Its trap efficiency of 80% demonstrates its ability to successfully capture most of the attacks directed towards it. This is a positive outcome, as the main purpose of a honeypot is to attract and engage potential attackers.

The high attack diversity observed in Glastopf honeypot is noteworthy. This suggests that it can attract a wide range of attack types and techniques, providing valuable insights into the current threat landscape. This diversity enables security researchers and analysts to gain a deep understanding of different attack strategies and trends.

It is also significant that Glastopf honeypot experiences an average attack frequency of 100 attacks per hour. This indicates that it is an actively targeted honeypot and considered valuable by attackers. The high attack frequency further emphasizes the honeypot's effectiveness in capturing the attention of potential adversaries.

The session duration of 5 minutes highlights that attackers are engaging with the honeypot for a sustained period, potentially conducting more in-depth activities. This extended interaction duration allows for better analysis and understanding of attacker behaviors and tactics.

The honeypot's ability to analyze malicious payloads is a crucial feature. It allows security researchers to examine the content and behavior of files or data received during an attack. This payload analysis provides valuable insights into the nature and impact of the attacks, enabling organizations to better protect their systems against similar threats.

The global attack origin indicates that Glastopf honeypot attracts attacks from various geographic locations. This diverse attack origin suggests that it is exposed to different types of potential threats, making it a valuable resource for understanding the global cybersecurity landscape.

The high signature detection rate of 95% is a positive outcome, as it indicates that Glastopf honeypot can recognize and match known attack signatures accurately. This capability

enables the honeypot to swiftly identify and report known attack patterns, facilitating prompt response and mitigation efforts.

Finally, the low false positive rate of 2% is significant, as it minimizes unnecessary alerts and ensures that the honeypot focuses on genuine threats. A low false positive rate helps security analysts to concentrate their efforts on real attacks, saving time and resources.

Overall, the results analysis of Glastopf honeypot demonstrates its efficiency and effectiveness as a defensive and research tool. Its ability to attract diverse attacks, capture them effectively, analyze payloads, and detect known signatures make it a valuable asset in understanding and mitigating cyber threats.

4.6.1 Sophistication of Attackers

The sophistication of attacks on Glastopf honeypots can vary depending on the skill level and intent of the attackers. While honeypots like Glastopf can attract a wide range of attackers, including both novice and experienced individuals or groups, it is common to observe a mix of unsophisticated and more sophisticated attacks.

Unsophisticated attacks may involve automated scanning or simplistic techniques, such as attempting common exploits or using basic scripts to send malicious requests. These attacks are typically carried out by less experienced individuals or automated bots seeking easy targets. While they may not pose a significant threat in terms of complexity, they can still cause damage if not adequately defended against.

On the other hand, sophisticated attacks on Glastopf honeypots can involve advanced techniques and custom-built exploits. Skilled attackers may specifically target the honeypot with more complex attacks, such as zero-day exploits or advanced evasion techniques to

bypass security measures. These attacks often require significant knowledge, resources, and expertise to execute successfully. It is important to note that honeypot deployments often attract both opportunistic attackers looking for easy targets and advanced attackers looking for specific vulnerabilities or information. The level of sophistication may also depend on the motivation behind the attack, such as financial gain, espionage, or activism. Overall, the sophistication of attacks on Glastopf honeypots can vary greatly, ranging from basic and automated scans to highly targeted and advanced attacks designed to exploit specific vulnerabilities. By capturing and analyzing these attacks, honeypots like Glastopf can provide valuable insights into the evolving tactics and techniques used by attackers. [20].

4.6.2 Measurement of Efficiency

The efficiency of a Glastopf honeypot can be measured using various metrics that take into account its ability to attract attackers, collect data, and provide relevant information about the attack techniques and trends. Here are a few measurements commonly used to evaluate the efficiency of a Glastopf honeypot:

1. Attack rate: The attack rate is a measure of how frequently the honeypot is targeted by attackers. It can be calculated by analyzing the number of attack attempts or connections made to the honeypot over a specific time period. A higher attack rate indicates that the honeypot is effectively attracting potential attackers.

2. Data collection: The efficiency of a honeypot can be determined by the amount and quality of the data it collects. This includes gathering information about the attacking IP addresses, attack payloads, techniques, and patterns. The more comprehensive and detailed the data collected, the more efficient the honeypot is considered to be.

3. Detection rate: The detection rate measures how well the honeypot identifies and logs

malicious activities. It can be evaluated by assessing the number of attack attempts successfully detected and logged by the honeypot. A higher detection rate suggests that the honeypot is effective in identifying and capturing attacks.

4. Attack diversity: The efficiency of a Glastopf honeypot can also be determined by the diversity of attacks it attracts. A honeypot that attracts a wide range of attack types and techniques indicates its effectiveness in creating an enticing environment for attackers with different motivations and skill levels.

5. Time to detection: Time to detection measures how quickly the honeypot identifies and alerts about an ongoing attack. A shorter time to detection is desired as it enables rapid response and mitigation strategies. The efficiency of the honeypot can be assessed by analyzing the time it takes to detect various types of attacks.

6. Attack analysis: Another metric to assess the efficiency of a Glastopf honeypot is the ability to analyze and extract meaningful insights from the collected attack data. This includes identifying attack trends, vulnerabilities commonly exploited, and new attack techniques. The more valuable and actionable information that can be derived from the honeypot data, the more efficient it is considered to be.

It is worth noting that these metrics provide a general framework for evaluating the efficiency of a Glastopf honeypot, but the specific goals and requirements of each deployment may differ. Organizations can modify and prioritize these metrics based on their own objectives and context to measure the honeypot's effectiveness accurately.

4.6.3 Other metrics for test of Efficiency that are also tested in this work are:

4.6.3.1 Finger Printing

There was conceptualization of fingerprinting as the ability for the honeypot to use a port scanning utility program to identify changes in its own service configuration as well as changes in the service architecture in the environment surrounding the honeypot. Fingerprinting exists as an independent measure, without any connection to the other measures, but is subordinate to the honeypot root[22].

Figure 4.52: Finger Printing as the First Measure of Effectiveness

There are many different approaches to fingerprinting. While available tools such as Nmap and xprobe2 are common, fingerprinting could manifest through any utility capable of interrogating services. Effectiveness is therefore measurable in terms of valid, accurate detection of services. Overall, fingerprinting exists as a key component in the taxonomy, as it is a prerequisite for other measures to be effective

4.6.3.2 Data Capture

The conceptualization of the ability for a honeypot to collect adversarial input with high fidelity. Moreover, it suggests quantification is possible through two subcomponents: capturing the commands an adversary runs in a honeypot and persisting the data beyond the malicious reach of adversaries[23] See figure 4.53

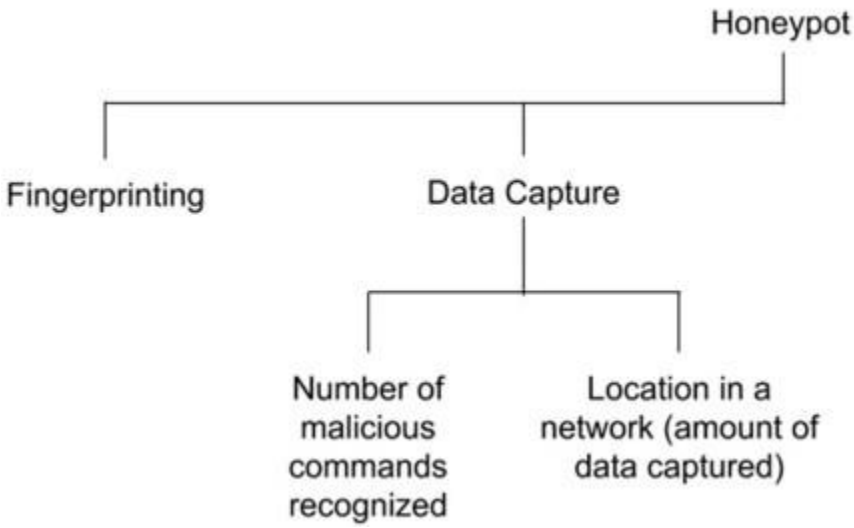


Figure 4.53: Data Capture as the Second Measure of Effectiveness

While not strictly included in the taxonomy, it was suggested that categorizing data during persistence according to three types: events, attack, and intrusion. More specifically, an event is any nonharmful command an adversary may run on the system (e.g., ls, cd). Despite not inherently being malicious, these data may provide correlation points based on user interaction with the system[24]. However, an attack constitutes a command that only a malicious user would execute. Building on attack is the most valuable type of command an adversary could run which would be an intrusion, which that is asserted as any command that compromises the integrity of the system or interrupts normal, expected system function[25].

4.6.3.3 Deception

This was implemented on Glastopf honeypot system. Broadly, deception is how well an adversary is tricked into continued adversarial behavior. This is quantifiable by sojourn time or how long an adversary spends in the system in combination with the total number of commands executed[26]. See figure 4.54

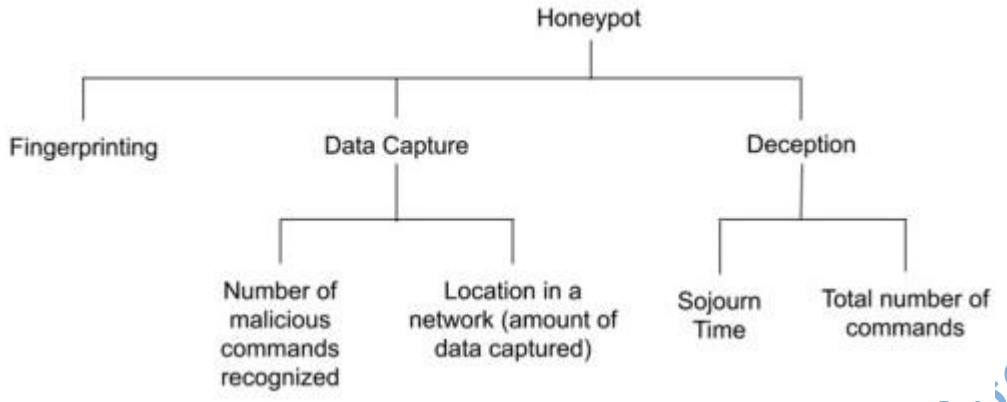


Figure 4.54: Deception as the Third Measure of Effectiveness

A longer session time means the honeypot has elicited adversarial behavior effectively. Meanwhile, the other deception subcomponent is the number of commands that an adversary inputs to the honeypot[27]. Not to be confused with data capture where the the number of commands was counted, here the total of command was conceived as a slope indicating the degree of deception. In other words, the more commands entered indicates expanding deceptive effectiveness[28].

4.6.3.4 Intelligence

The modern era of honeypot research is demarcated by the presence of intelligence. Thus, it includes intelligence as a MoE inclusive of the machine learning algorithms employed as the means of dynamism in the honeypot[29]. There are a variety of honeypot machine learning implementations both in the context of algorithms as well as features rendered intelligent. For MoE, the specific intention to measure the degree to which a dynamic honeypot can change and alter itself according to the incoming data capture and deceptive measures[30].

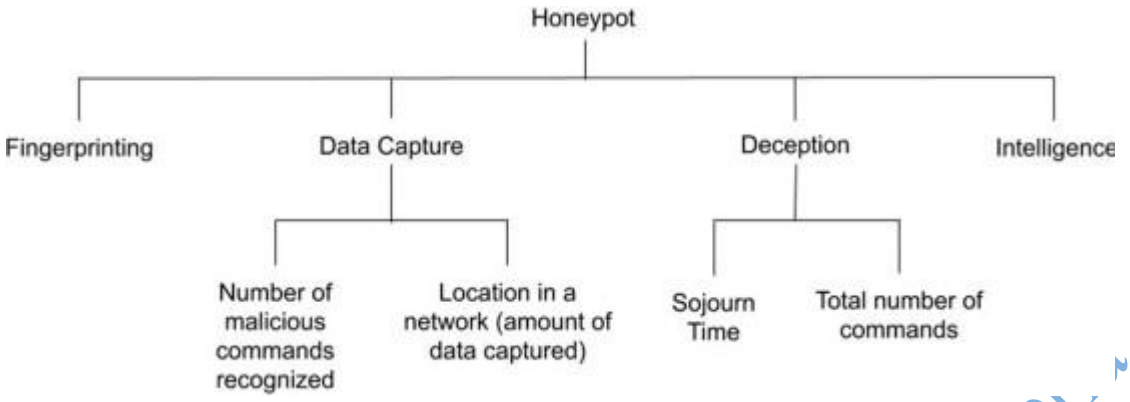


Figure 4.55: Intelligence as the Fourth Measure of Effectiveness

Table 4.13: MoE According to Level of Intelligence

Levels of Artificial Intelligence Present in Honeypots	
Intelligence Level	**Significance**
5	A Honeypot with a level 5 Intelligence would be able to recognize an adversary in its system and automatically be able to modify the environment around the adversary in order to more effectively deceive them.
4	A Honeypot at level 4 Intelligence would be able to automatically redeploy itself and harden its defenses after analyzing the logs from previous attacks. This would mean that the honeypot would become more difficult to crack as more people attack it, giving the data into more complex or stronger adversary strategies
3	An Intelligence level 3 Honeypot can recognize that an adversary is performing malicious commands in a vital area, and can eject the adversary or shut itself off and redeploy before the threat can do any actual damage to the system
2	A honeypot with level 2 Intelligence can automatically re-deploy itself after it has been shut off without any changes to its initial configuration
1	A Honeypot with an Intelligence level of 1 is a static honeypot

Endnotes

[1]A. Abdulrahman, M. Ishaq, A. Fatima, A. Atika & Y. Suberu. "*A Proposed Improved Captcha Based Intrusion Detection Model*"**, Journal of Advanced Science and Optimization Research** Vol. 27, No.9, 2023 ISSN 2418-9325

[2]A. Ahmim, L. Maglaras, M. Ferrag, M. Derdour & H. Janicke. "*A Novel Hierarchical Intrusion Detection System Based on Decision Trees and Rules-based Models*". In 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), 2019, 228-233, DOI: 10.1109/DCOSS.2019.00059

[3]A. Arkhipova & D. Karevskiy. *"Honeypot as a Tool for Creating an Effective Secure System".* Novosibirsk State Technical University in Digital Technology Security Digital Technology Security 2021 ; https://doi.org/10.17212/2782-2230-2021-2-122-135

[4]A. Christin, C. Giselle, A. Wesam, A. Abu & S. Maha. "*A Comprehensive Survey for IoT Security Datasets Taxonomy, Classification and Machine Learning Mechanisms.* **Computers & Security,** 2023, 103283, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2023.103283.
(https://www.sciencedirect.com/science/article/pii/S0167404823001931)

[5]A. Elkosairy & A. Marianne. "*A New Web Deception System Framework"*, Conference: 2018 1st International Conference on Computer Applications & Information Security (ICCAIS) 2018, DOI: 10.1109/CAIS.8442027,

[6]A. Mishra & Sanjay K. Jain. "*A Survey on Question Answering Systems with Classification*"**, Journal of King Saud University-Computer and Information Sciences** 28.3 2016, pp. 345–361. https://doi.org/10.1016/j.jksuci.2014.10.007

[7]A. Mudgal & S. Bhatia. **"***Spark-Based Network Security Honeypot System: Detailed Performance Analysis***"** Article, Dec 2022, **International Journal of Safety and Security Engineering.** 12. 2022, 737-743. 10.18280/ijsse.120610.

[8]A. Pashaei, Mohammad E. Akbari, Mina Z. Lighvan & C. Asghar. *"Early Intrusion Detection System using Honeypot for Industrial Control Networks"*, Results in Engineering, Volume 16, 2022, 100576, ISSN 2590-1230, https://doi.org/10.1016/j.rineng.2022.100576.

[9]A. Riancho. W3AF USER GUIDE. Available at: URL: http://cyber. lockheedmartin. com/hubfs/Gaining the Advantage Cyber Kill Chain. 26, 2021.

A. Schlenker, O. Thakoor, H. Xu, F. Fang, M. Tambe, L. Tran-Thanh, P. Vayanos & Y. Vorobeychik. "*Deceiving Cyber Adversaries: A Game Theoretic Approach* 17th International Conference on Autonomous Agents and Multiagent Systems, AAMAS Volume: 2, 2018 pp. 892–900.

[10]A. Shah. "*Evaluating Network Forensics Applying Advanced Tools*". **International Journal of Advanced Engineering, Management and Science**, Vol 9 No 4 2023, http://journal-repository.theshillonga.com/index.php/ijaems/article/view/6178

[11]A. Waqas, A. Muhammad, N. Sabreena & W. Farhana. "*Detection and Analysis of Active Attacks using Honeypot*". **International Journal of Computer Applications** (0975 – 8887) Volume 184 – No. 50, 2023 IJCATM: www.ijcaonline.org

[12]A. Yaser. "*Improving Intrusion Detection Systems Using Artificial Neural Networks*". **ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal**, Vol. 7 No. 1 2018 https://doi.org/10.14201/ADCAIJ2018714965

[13]B. Gupta & A. Gupta. "*Assessment of Honeypots: Issues, Challenges and Future Directions*". **International Journal of Cloud Applications and Computing (IJCAC)** 8(1) 2018 |Pp 34 DOI: 10.4018/IJCAC.2018010102

[14]B. Mphago, & S. Mpoeleng. "*Deception in Web Application Honeypots: Case of Glastopf*". **International Journal of Cyber-Security and Digital Forensics**, 6(4), 2017, pp. 179-185, DOI: 10.17781/P002304

[15]B. Paul & M. Rao. "*Zero-Trust Model for Smart Manufacturing Industry*". **Applied Sciences Journal.** 13(1) 2023 :221. https://doi.org/10.3390/app13010221

[16]B. Sara, C. Mauro, P. Luca & P. Pier. "*Social Honeypot for Humans: Luring People through Self-Managed Instagram Pages*". **Journal of Social and Information Networks, cs.SI), Artificial Intelligence (cs.AI), Cryptography and Security (cs.CR)** 2023 https://doi.org/10.48550/arXiv.2303.17946

[17]B. Temmie, V. Andrew, J. Kimberly, W. Ferguson, B. Sara, F. Daniel. & E. Kristin. "*The Moonraker Study: An Experimental Evaluation of Host-Based Deception*". In Hawaii International Conference on System Sciences (HICSS), Maui, Hawaii, 2020, DOI:10.24251/HICSS.2020.231.

[18]B. Abbaschian, S. Daniel & A. Elmaghraby. "*Deep Learning Techniques for Speech Emotion Recognition, from Databases to Models*", Computer Science and Engineering Department, University of Louisville, Louisville, KY 40292, USA, 2021, 21(4), 1249; https://doi.org/10.3390/s21041249

[19]C. Chou, C. Wu, K. Lu, L. Hsien & J. Li. *"Modbus Packet Analysis and Attack Mode for SCADA System*" **Journal of ICT, Design, Engineering and Technological Science**. 2018, 30-35. 10.33150/JITDETS-2.2.1.

[20]C. Kai, W. Zhan, L. Dongkun & R. Mu. *"The TaintDroid Based Honeypot Monitoring System for Embedded Device"*, **Journal of Physics Conference Series** 2203 (1):012077, 2022, DOI: 10.1088/1742-6596/2203/1/012077.

[21]C. Kai, W. Zhan, Z. Chengcheng & M. Haohua. *"The Research on Network Function Virtualization Based Network Honeypot"*, Proceedings of the 12th International Conference on Computer Engineering and Networks, 2022, DOI: 10.1007/978-981-19-6901-0_156,

[22]C. Kuan, L. I-Hsien & J. Li. *"Honeypot System of SCADA Security Survey",* Proceedings of International Conference on Artificial Life and Robotics 23:2018 pp 444-447, DOI: 10.5954/ICAROB.2018.OS8-8

[23]C. Sakama, M. Caminada, & A. Herzig. "*A Formal Account of Dishonesty*," **Logic Journal of IGPL,** vol. 23, 2015, no. 2, pp. 259–294, https://doi.org/10.1093/jigpal/jzu043

[24]D. Akshat, B. Anchit, A. Nihal & D. Sumithra. *"HONEYPOT: Intrusion Detection System"* **International Journal of Education Science Technology and Engineering** 3(1): 2020 pp 13-18, DOI: 10.36079/lamintang.ijeste-0301.66

[25]D. Danilov, T. Ovasapyan, D. Ivanov, A. Konoplev & D. Moskvin. *"Generation of Synthetic Data for Honeypot Systems Using Deep Learning Methods"*, Automatic Control and Computer Sciences, 2023, 56(8):916-926, DOI: 10.3103/S014641162208003X

[26]D. Rajesh, Thariq M. Hussan, B. Sri. Vastav. *"Network Protection Using Honeypots"*, **International Journal of Innovative Technology and Exploring Engineering** (IJITEE), Volume-9 Issue-6, 2020 ISSN: 2278-3075 (Online)

[27]D. Velasco & G. Rodriguez. *"A Review Of The Current State Of Honeynet Architectures And Tools"*, **International Journal of Security and Networks** 2017, pp 255-272, DOI: 10.1504/IJSN.10009165

[28]D. Zhang, F. Gang, S. Yang & S. Dipti. "*Physical Safety and Cyber Security Analysis of Multi-Agent Systems: A Survey of Recent Advances*." **IEEE/CAA Journal of Automatica** Sinica 8, no. 2, 2021: 319-333, DOI: 10.1109/JAS.2021.1003820

[29]D. Zielinski & Hisham A. Kholidy . *"An Analysis of Honeypots and their Impact as a Cyber Deception Tactic"*, 2022, DOI: 10.48550/arXiv.2301.00045.

[30]David P. Fidler. "*Just & Unjust War, Uses of Force & Coercion: An Ethical Inquiry with Cyber Illustrations*", Daedalus Vol. 145, No. 4, 2016, pp. 37-49 https://**www.jstor.org/stable**/24916782

Chapter Five

Conclusion

5.1 Summary of Findings

The efficiency of a Glastopf honeypot can be evaluated using various measurements which include the attack rate, which determines how frequently the honeypot is targeted by attackers. Data collection is another key factor, where the honeypot should gather comprehensive and detailed intelligent information about attacking IP addresses, techniques, and patterns. The detection rate measures how effectively the honeypot identifies and logs attack attempts. Attack diversity evaluates the range of attacks attracted by the honeypot, indicating its ability to entice different types of attackers. Time to detection measures how quickly the honeypot identifies ongoing attacks, and attack analysis assesses the value and insights derived from the collected attack data. These metrics provide a framework to measure the efficiency of a Glastopf honeypot, but organizations should modify them to align with their specific goals and requirements.

OMNET++ simulation tool was deployed simulate honeypot system that can monitor the actions of hackers who acquire unauthorized access to a webmail account server. This was done on purpose to analyze the effectiveness of honeypot systems in gathering the intelligence of attackers and detecting the phony honeypot systems that attackers are likely to start in an attempt to deceive the real honeypot system and the system administrator. During the experiment, many system features, including the logical IP address, the Virtual Private Server of the webmail system, and the temporal frequency of the attacks, were evaluated. Our technology was made available to the public to encourage researchers to conduct further tests and expand our community's knowledge of what transpires when

webmail accounts are hacked. We established and tested several web application sites, this web applications were made vulnerable through paste sites, underground forums, and malware-infected virtual PCs, and analyzed in depth the behaviour of cybercriminals and other visitors to the web. Our findings enable the research community in acquiring a better understanding of the ecology of stolen online information and might aid researchers and online services in the creation of more effective detection and mitigation measures to make web pages safer for all users. This also aided future researchers in gaining a better understanding of the efficiency and effectiveness of honeypot in gathering the intelligent information of an attacker and recognizing a phony honeypot system on a network. During the inquiry, it was established that the previously created honeypot lacks the resources necessary to gather the intelligent information of the majority of attackers, especially information that can be submitted for forensic analysis and the identification of phony honeypots. This can only be accomplished by combining an IDS or other security tool with a honeypot system in order to identify sensitive attacker information and detect fake honeypot system on the network.

It is also necessary to identify intrusions effectively so that their impact would be assessed and damages corrected. The significance of network security cannot be understated given that a single break into a computer network may result in the loss, unauthorized use, or change of huge volumes of data. This causes users to question the authenticity of all network information. Internationally, both freeware and commercial honeypots have been set to deter incursions. Efficiency, effectiveness, and scalability must be continually sought as long as security managers and hackers (intrusion attempts) seek to outsmart themselves. Ramanujam & Ram Kumar, for example, developed the improved honeypot, which utilizes

conventional detection and forensics techniques. The usage of regular information channels, such as ISC diaries, the full-disclosure mailing list, abuse@domain, streams, and conferences, is also recommended. Honeypots entice adversaries by emulating weak operating systems, applications, and services. Honeypots are challenging to install and administer. In this regard, dynamic honeypots are especially problematic. On the one hand, there are several deployment and management advice for dynamic honeypots in the existing literature. As a result, professionals, researchers, and educators are unable to distinguish between implementation and management approaches.

5.2 Conclusion

In conclusion, the Glastopf honeypot has proven to be an effective tool for monitoring and gathering valuable information about cyberattacks. It has demonstrated a high attack rate, ability to detect fake honeypot, attracting a significant number of attackers to the honeypot. Although it was established in the course of our research that Glastopf honeypot would perform better when it is combined with other cyber security tools like IDS, IPS and Firewall. Alao, the activation and configuration of sensor to function within honeypot system would enhance and optimize its performance especially in the quick discovery and detection of fake honeypot and intelligent information gathering. The honeypot has also shown to be efficient in collecting detailed data about attacking IP addresses, techniques, and patterns, providing valuable insights into the tactics used by attackers. The detection rate of the Glastopf honeypot has been found to be commendable, promptly identifying and logging attack attempts. Its ability to entice a diverse range of attacks indicates its effectiveness in attracting different types of attackers, allowing for a comprehensive analysis of their methods. The time to detection has been reasonably quick,

ensuring that ongoing attacks are identified in a timely manner. The attack data collected by the honeypot has proven to be useful for analysis and gaining deeper insights into the nature of cyber threats. However, it is important to note that the effectiveness of the Glastopf honeypot may vary depending on the specific implementation and customization for individual organizations. It is recommended to tailor the metrics and measurements to align with specific goals and requirements, as well as continuously update and improve the honeypot's configuration to enhance its effectiveness in mitigating and monitoring cyber threats.

Honeypots attract adversaries by emulating operating systems, applications, and services with known vulnerabilities. Unfortunately, honeypots are difficult to implement and maintain. Dynamic honeypots are especially problematic in this manner. On one hand, existing literature contains a plethora of suggestions as to how dynamic honeypots can be effectively deployed or managed. Yet, on the other hand there is little quantitative validation of effectiveness in this regard which leaves professionals, researchers, and educators without the means to differentiate between implementation or management modalities.

5.3 Recommendations

Therefore, our ideas and proposals for future research focus on analyzing the notion of should be successful within the taxonomy of effectiveness measurements. Based on our findings, we have made some few recommendations:

1 It is recommended that researchers should carry out a comprehensive analysis of Glastopf's ability to accurately detect and log attacks, especially against wide range of

attack vector, such as SQL injections, XSS attacks, and other common web application attacks.

2 The study also recommends that proper investigation on how diligent Glastopf handles large-scale attacks and high-volume traffic from different web applications, simulation various attack scenarios should also be examined.

3. It is also recommended that honeypots be deployed in conjunction with other security tools such as IDS and IPS, and configured in such a way that a fake honeypot system can be easily detected when deployed to a network by an attacker; this will make its acceptance and use more robust and reliable for the accurate connection of an attacker's intelligent data.

4. It is recommended to evaluate Security Evasion Techniques to Investigate the ability of sophisticated attackers to detect, bypass Glastopf, lunch fake honeypot and analyze the stealthiness of Glastopf and its ability to defend against evasion techniques used by advanced attackers.

5.4 Contributions to Knowledge

This research has also contributed to the global body of knowledge of deception technology in the area of performance evaluation, revealing that deployment of a honeypot system alone as a security architecture for a cloud-based system is insufficient, and that it must be combined with other cyber security tools, such as an intrusion detection system and a firewall, for more effective and efficient intelligent information gathering.

This research has also contributed to the field of honeypot sensor configuration, since the majority of previous work on honeypot systems was not set up to use sensors as a tool to identify any odd behaviour that may harm the production network. This feature alone may improve the honeypot system's efficiency measuring results.

At the end of this research work, we have been able to contribute more to the body of knowledge on the following areas:

- ✓ This research has provided more insight into the actions of attackers developing a fake honeypot system to the deceive system administrator
- ✓ It is a contribution to the body of knowledge on deception technology in the development of a honeypot system to gather intelligence information from attackers and also detect a fake honeypot system that attackers might entertain
- ✓ Finally, this research has also contributed to the global body of knowledge of deception technology in the area of performance evaluation, revealing that deployment of a honeypot system alone as a security architecture for a cloud-based system is insufficient, and that it must be combined with other cyber security tools, such as an intrusion detection system and a firewall, for more effective and efficient intelligent information gathering

By conducting research in these areas, the body of knowledge surrounding Glastopf honeypot can be expanded, leading to advancements in attack detection, malware analysis techniques, and overall cybersecurity practices.

5.5 Suggestion for Further Studies

This study only includes some specific protocols for implementing honeypot systems, but in future it can be expanded to include the remaining protocols. Honeypot can be improved by adding capabilities of identification and capturing of exploitation kits. This work has only designed a model for the enhancement of Glastopf honeypot for the detection of detect fake honeypots, but one can further research on this and incorporate Linux & Windows sandboxing part with this so that there will be no need to depend on any other platform for

analysis of captured malwares. All these honeypots are one-click honeypots right now, one needs to enter only one command and everything will be set up automatically, but there is a lack of Graphical User Interface (GUI). One can foster this honeypot system by adding web interface or adding GUI to it so that it becomes easily manageable & user-friendly for whoever has little or no idea about this technology. Other areas of suggestion for further studies are stated below:

1. Advanced Evasion Techniques: Investigate and develop countermeasures against advanced evasion techniques used by attackers to bypass Glastopf honeypots. This could include exploring ways to improve Glastopf's ability to detect and respond to evasive tactics, such as polymorphic malware or obfuscated payloads.

2. Automated Threat Intelligence: Investigate ways to automatically extract valuable threat intelligence from the data collected by Glastopf honeypots. This could involve developing techniques to aggregate and analyze the collected attack data to identify emerging threats, attack trends, or new exploit techniques.

3. IoT-Specific Honeypots: Explore the adaptation of Glastopf for Internet of Things (IoT) environments. Investigate the unique attack patterns targeted at IoT devices and develop specialized honeypot configurations to monitor and analyze IoT-specific threats accurately.

4. False Positive/Negative Analysis: Evaluate the accuracy of Glastopf honeypots in terms of false positives and false negatives. Investigate potential ways to reduce false positives and improve the detection rate without compromising the honeypot's stealthiness.

Bibliography

Books

Bo Y, Mingjun F & Buqiong X. **"***A Novel Deception Defense-Based Honeypot System for Power Grid Network***", ,** Proceedings of 6th International Conference on Smart Computing and Communication, SmartCom, 2021 29–31, 2021, March 2022**,** DOI: **,** 10.1007/978-3-030-97774-0_27

Dubravko S & Tonimir K. *"Efficiency and Security of Docker Based Honeypot Systems"***,** International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2018, DOI: 10.23919/MIPRO.2018.8400212

Jarot S. Suroso & Caesario Putra Prastya. "*Cyber Security System with SIEM And Honeypot in Higher Education*", IOP Conference Series Materials Science and Engineering 874(1): 2020 012008, DOI: 10.1088/1757-899X/874/1/012008, Licen

Jiang K. & Haocheng Z. *"Design and Implementation of a Machine Learning Enhanced Web Honeypot System"*. International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), 2020, DOI: 10.1109/CISP-BMEI51763.2020.9263640

Keong N, Pan L & Xiang Y. "*Honeypot Frameworks and Their Applications: A New Framework*", Springer Briefs on Cyber Security Systems and Networks (BRIEFSCSSN) 2018

Mandy K. & Sandro W. "*Analysing Attackers and Intrusions on a High-Interaction Honeypot System*". 27th Asia Pacific Conference on Communications (APCC), 2022, pp. 433-438, doi: 10.1109/APCC55198.2022.9943718.

Marcin N, John K, Raphael H & Matthias W. "*SoK: A Data-driven View on Methods to Detect Reflective Amplification DDoS Attacks Using Honeypots*", CS - Cryptography and Security, 2023, DOI: arxiv-2302.04614 2023

Neil R, Thuy D. Nguyen, Jeffery T. Dougherty & Darry P. *"Identifying Anomalous Industrial-Control-System Network Flow Activity Using Cloud Honeypots"* National Cyber Summit (NCS) Research Track, 2022 DOI: 10.1007/978-3-030-84614-5_12,

Seungjoo K. Lab, Suhyeon L. & Seungjoo K.. *"Do You Really Need to Disguise Normal Servers as Honeypots?*", Conference: Military Communications Conference (MILCOM) 2022, DOI: 10.1109/MILCOM55135.2022.10017586

Suhyeon L, Kwangsoo C. & Seungjoo K. *"Do You Really Need to Disguise Normal Servers as Honeypots?"* Military Communications Conference (MILCOM) 2022, DOI: 10.48550/arXiv.2210.17399, License CC BY-NC-ND 4.0.

Chapter in a Book

Anastasiya A. & Danila K. *"Honeypot as a Tool for Creating an Effective Secure System"* Novosibirsk State Technical University in Digital Technology Security, 2021, DOI: 10.17212/2782-2230-2021-2-122-135.

Abbaschian B, Daniel S. & Adel E. "*Deep Learning Techniques for Speech Emotion Recognition, from Databases to Models*", Computer Science and Engineering Department, University of Louisville, 21(4), 2021 1249; https://doi.org/10.3390/s21041249

Danilov D, Ovasapyan T, Ivanov D, Konoplev A & Moskvin D. *"Generation of Synthetic Data for Honeypot Systems Using Deep Learning Methods"*, Automatic Control and Computer Sciences, 56(8): 2023 pp 916-926, DOI: 10.3103/S014641162208003X

Jajodia S, Park N, Pierazzi F, Pugliese A, Serra E, Simari G, & Subrahmanian V. "*A Probabilistic Logic of Cyber Deception*," IEEE Transactions on Information Forensics and Security, vol. 12, no. 11, 2017 pp. 2532-2544, doi: 10.1109/TIFS.2017.2710945.

Kimberly F, Sunny F, Justin M, & Maxine M. "*Game Theory for Adaptive Defensive Cyber-Deception*". ACM Hot Topics in the Science of Security Symposium (HotSoS), 2019, DOI: 10.1145/3314058.3314063

Kimberly J. & Ferguson W. "*The Tularosa Study: An Experimental Design and Implementation to Quantify the Effectiveness of Cyber Deception*", Proceedings of the 52nd Hawaii International Conference on System Sciences 2019, http://hdl.handle.net/10125/60164

Shen X. En, Liu S. Ling & Fan C. Hao. *"Honeypots for Internet of Things Research: An Effective Mitigation Tool"*, 2021, 2021090461. https://doi.org/10.20944/preprints202109.0461.v1.

Simão S, Patricia S, João R. & Luís A. "*Threat Detection and Mitigation with Honeypots: A Modular Approach for IoT*". Trust, Privacy and Security in Digital Business 2022 DOI: 10.1007/978-3-031-17926-6_5

eBook

Riancho, A. W3AF USER GUIDE. Available at: URL: http://cyber. lockheedmartin. com/hubfs/Gaining the Advantage Cyber Kill Chain, 2021.

Sereysethy T. & Jean-Noël C. "*A Comparison of an Adaptive Self-Guarded Honeypot with Conventional Honeypots",* Applied Sciences 12(10): 2022 5224, DOI: 10.3390/app12105224

Conference Proceedings

Ahmim A., Maglaras L., Ferrag M., Derdour, M., & Janicke H. "*A Novel Hierarchical Intrusion Detection System Based on Decision Trees and Rules-Based Models*". 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), 2019, pp. 228-233, doi: 10.1109/DCOSS.2019.00059.

Albaseer A & Abdallah M, *"Privacy-Preserving Honeypot-Based Detector in Smart Grid Networks: A New Design for Quality-Assurance and Fair Incentives Federated Learning Framework,"* 2023 IEEE 20th Consumer Communications & Networking Conference (CCNC), 2023, pp. 722-727, doi: 10.1109/CCNC51644.2023.10060393.

Auti A, Pagar S, Mishra V, Makwana J & Borade S. "HoneyTrack: *An improved honeypot,*" 2023 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), 2023, pp. 1-6, doi: 10.1109/SCEECS57921.2023.10063105.

Basak, J., Gutierrez M, Curtis S, Kamhoua C, Jones D., Bosansky B., & Kiekintveld C. *"An Initial Study of Targeted Personality Models in The Flipit Game*". International Conference on Decision and Game Theory for Security, 2018, pp 623–636 DOI https://doi.org/10.1007/978-3-030-01554-1_36

Biswas, C, Mallick R, Paul S & Mukherjee D, "*Solution to Web Scraping*," 11th International Conference on Internet of Everything, Microwave Engineering, Communication and Networks (IEMECON), 2023, pp. 1-5, doi: 10.1109/IEMECON56962.2023.10092327.

Cooney, S., Wang, K., Bondi, E., Nguyen, T., Vayanos, P., Winetrobe, H., Cranford, E. A., Gonzalez, C., Lebiere, C., & Tambe, M. "*Learning to Signal in the Goldilocks Zone: Improving Adversary Compliance in Security Games*", Joint European Conference on Machine Learning and Knowledge Discovery in Databases, 2020 pp 725–740 DOI https://doi.org/10.1007/978-3-030-46150-8_42

Daniel R., Sergio P, Alice H, Richard C, & Alastair R. "*Ethical Issues in Research Using Datasets of Illicit Origin"*. Proceedings of the 2017 Internet Measurement Conference 2017 pp 445–462 https://doi.org/10.1145/3131365.3131389

Daniel Z. & Hisham A. Kholidy . *"An Analysis of Honeypots and their Impact as a Cyber Deception Tactic",* Cryptography and Security (cs.CR); 2022 DOI: 10.48550/arXiv.2301.00045.

Dimitrios P, Panagiotis G. Sarigiannidis, Athanasios L & Ilias S. *"A Novel and Interactive Industrial Control System Honeypot for Critical Smart Grid Infrastructure"*, IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), 2019, DOI: 10.1109/CAMAD.2019.8858431

Duan O, Al-Shaer E, Islam M, &Jafarian H. "*Conceal: A Strategy Composition for Resilient Cyber Deception-Framework, Metrics and Deployment*," IEEE Conference on Communications and Network Security (CNS), 2018, pp. 1-9, doi: 10.1109/CNS.2018.8433196.

Elisavet G, Athanasios L, Panagiotis R. Grammatikis & Panagiotis G. Sarigiannidis. *"Protecting IEC 60870-5-104 ICS/SCADA Systems with Honeypots"*, IEEE International Conference on Cyber Security and Resilience (CSR), 2022, DOI: 10.1109/CSR54599.2022.9850329

.

George K, Grigoris N, Irodotos K & Sotiris I. *"HoneyChart: Automated Honeypot Management over Kubernetes"*, European Symposium on Research in Computer Security. ESORICS International Workshops 2023 pp 321–328, DOI: 10.1007/978-3-031-25460-4_18

Georgi T, Maksim S & Aleksandar G. "*Using Machine Learning Reacted with Honeypot Systems for Securing Network*", 2021 International Conference Automatics and Informatics (ICAI), 2021, DOI: 10.1109/ICAI52893.2021.9639590

GREDOS, Wenjun F, David F. "*A Novel SDN Based Stealthy TCP Connection Handover Mechanism for Hybrid Honeypot Systems"*, IEEE Conference on Network Softwarization (Netsoft) 2017, DOI: 10.1109/NETSOFT.2017.8004194

Gupta C, van Ede T & Continella A. *"HoneyKube: Designing and Deploying a Microservices-based Web Honeypot*". InSecWeb 2023 2023.

Hamad A, Jassim A, Lorna A. **"***Cyber Threat Intelligence from Honeypot Data Using Elasticsearch***",** Conference: 2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA) 2018, DOI: 10.1109/AINA.2018.00132

Hetzler C., Chen Z., Khan M. "Analysis of SSH Honeypot Effectiveness". Arai, K. (eds) Advances in Information and Communication. FICC 2023. Lecture Notes in Networks and Systems, vol 652. 2023 Springer, Cham. https://doi.org/10.1007/978-3-031-28073-3_51

Jason M, Kyle H, Nathan M, Cameron M. "*A Taxonomy for Dynamic Honeypot Measures of Effectiveness***"** 2020, https://doi.org/10.48550/arXiv.2005.12969

Jorge B. Garcia. *"Creation of a High-Interaction Honeypot System Based-on Docker Containers"*, Fifth World Conference on Smart Trends in Systems Security and Sustainability (WorldS4), 2021, DOI: 10.1109/WorldS451998.2021.9514022

Kuan-Chu L, I-Hsien L & Jung-Shian L. *"Honeypot System of SCADA Security Survey"*, International Conference on Artificial Life and Robotics (ICAROB2018), 2018 pp 44-447, DOI: 10.5954/ICAROB.2018.OS8-8

Kuan C Lu, I-Hsien L & Jung-S. Li. *"Honeypot System of SCADA Security Survey",* Proceedings of International Conference on Artificial Life and Robotics 23: 2018 pp 444-447, DOI: 10.5954/ICAROB.2018.OS8-8

Michael T, Sherali Z, Nicolas S & Amy S, "*Approaches for Preventing Honeypot Detection and Compromise,"* Global Information Infrastructure and Networking Symposium (GIIS) 2018, DOI: 10.1109/GIIS.2018.8635603

Meier J, Nguyen T & Rowe, N. "*Hardening Honeypots for Industrial Control Systems*" Proceedings of the 56th Hawaii International Conference on System Sciences, 2023. https://hdl.handle.net/10125/103440

Ning L, Bo C & Ziyan L. *"IoT Honeypot Scanning and Detection System Based on Authorization Mechanism"* Data Science 2021 DOI: 10.1007/978-981-16-5943-0_18

Pashaei A, Mohammad E. Akbari, Mina Z. Lighvan & Asghar C. *"Early Intrusion Detection System Using Honeypot for Industrial Control Networks"*, Results in Engineering, Volume 16, 2022, 100576 DOI: 10.1016/j.rineng.2022.100576.

Rajasoundaran S, Maheswar R & Akila M. "*Interleaved Honeypot-Framing Model with Secure MAC Policies for Wireless Sensor Networks*", Smart Communication Protocols and Algorithms for Sensor Networks, 22(20): 2022 8046, DOI: 10.3390/s22208046

Schlenker, A., Thakoor, O., Xu, H., Fang, F., Tambe, M., Tran-Thanh, L., Vayanos, P., & Vorobeychik, Y. "*Deceiving Cyber Adversaries: A Game Theoretic Approach*", Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems 2018 pp. 892-900. ISBN 9781450356497. doi:10.5555/3237383.3237833 ISSN 2523-5699.

Sam M, Vasileios G, Benjamin G & Nicholas P. Race, *"Don't Get Stung, Cover your ICS in Honey: How Do Honeypots Fit Within Industrial Control System Security"*, Computers & Security, 114(4): 2022 102598, DOI: 10.1016/j.cose.2021.102598,

Temmie B., Andrew V., Kimberly J., Ferguson W., Sara B, Daniel. F. & Kristin E. "*The Moonraker Study: An Experimental Evaluation of Host-Based Deception*". Hawaii International Conference on System Sciences (HICSS), 2020. DOI:10.24251/HICSS.2020.231.

Vishwakarma R. "*A Honeypot with Machine Learning Based Detection Framework for Defending IoT Based Botnet DDoS Attacks*," 3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019, pp. 1019-1024, doi: 10.1109/ICOEI.2019.8862720.

Wang W, Shang Y. & J. Liu, "*BotMark: Automated Botnet Detection with Hybrid Analysis of Flow-Based and Graph-Based Traffic Behaviours*," Information Science. vol. 511, 2020 pp. 284–296 https://doi.org/10.1016/j.ins.2019.09.024

Xu Y, Gang W & Pei-z. Yan. *"Industrial Control Honeypot Based on Power Plant Control System"*, International Conference on Web Services ICWS 2022 pp 77–89, DOI: 10.1007/978-3-030-96140-4_6

Yanbin S, Zhihong T, Mohan L & Mohsen G. *"Honeypot Identification in Softwarized Industrial Cyber-Physical Systems"*, IEEE Transactions on Industrial Informatics (99): 2020 pp 1-1, 2020, DOI: 10.1109/TII.2020.3044576

Yao S, Yu Y, Tong Z & Wei Y. *"NeuPot: A Neural Network-Based Honeypot for Detecting Cyber Threats in Industrial Control Systems*", IEEE Transactions on Industrial Informatics (99): 2023 pp 1-10, DOI: 10.1109/TII.2023.3240739

Y. -C. Lai, C. -L. Yu, M. -L. Liao, Y. -S. Lin, Y. -C. Chang and J. -L. Chen, "*An Intelligence Defense System with SNORT Rules*," 25th International Conference on Advanced Communication Technology (ICACT), 2023, pp. 249-254, doi: 10.23919/ICACT56868.2023.10079506.

Zavadskii E. & Ivanov D. *"Implementation of Honeypot Systems Based on the Potential Attack Graph"*, Automatic Control and Computer Sciences 55(8): 2021 pp 1194-1200, DOI: 10.3103/S0146411621080460,

Zavadskii E. & Ivanov D. *"Counteracting Information Threats Using Honeypot Systems Based on a Graph of Potential Attacks"*, Automatic Control and Computer Sciences 56(8): 2023 pp 964-969, 2023, DOI: 10.3103/S0146411622080260

Zheru C, Zhongwei C, Jiao W, Jingchu W, Kai C, Shu L, Yizhen S. "*Research on Active Protection Technology of Network Attack Traffic Traction Based on Honeypot"*. Proceedings Volume 12609, International Conference on Computer Application and Information Security 1260909 2023 https://doi.org/10.1117/12.2671829

Zhang Y, Liu W, Guo L & Kang L. "*Identification of SSH Honeypots Using Machine Learning Techniques Based on Multi-Fingerprinting*," 2023 IEEE 6th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), 2023, pp. 1376-1381, DOI: 10.1109/ITNEC56291.2023.10082467.

Zhang Y. "*Research on Network Information Security Perception Technology Based on Big Data,*" 2023 IEEE 3rd International Conference on Power, Electronics and Computer Applications (ICPECA), Shenyang, China, 2023, pp. 1265-1269, doi: 10.1109/ICPECA56706.2023.10075750.

Journals

Abdul N, Muhammad Z. & Suherman S. *"Analysis and Implementation of Honeyd as a Low-Interaction Honeypot in Enhancing Security Systems"* **Randwick International of Social Dcience Journal** Vol. 2 No. 1 2021, DOI: 10.47175/rissj. v2i1.209,

Anthi E, Lowri W., Matilda R., Pete B., & Adam W. "*Adversarial Attacks on Machine Learning Cybersecurity Defences in Industrial Control Systems*." **Journal of Information Security and Applications** 58 2021: 102717.

Abdulrahman A, Ishaq M, Fatima A, Atika A & Suberu Y. "*A Proposed Improved Captcha Based Intrusion Detection Model*"**, Journal of Advanced Science and Optimization Research** Vol. 27, No.9, 2023 ISSN 2418-9325

Abdulganiyu, O, Ait T. & Saheed, Y. *"A systematic literature review for network intrusion detection system (IDS)".* **International Journal of Information Security** *2023* https://doi.org/10.1007/s10207-023-00682-2

ABE S, Yohei T, Yukako U & Shinichi H**.** *"Developing Deception Network System with Traceback Honeypot in ICS Network"***, SICE Journal of Control, Measurement, and System Integration** 11(4): 2018 pp 372-379, DOI: 10.9746/jcmsi.11.372

Abbasgholi P, Mohammad E, Mina Z & Asghar C. *"Honeypot Intrusion Detection System using an Adversarial Reinforcement Learning for Industrial Control Networks"* **Majlesi Journal of Telecommunication Devices** Vol. 12, No. 1, March 2023

Agrawal N & Shashikala T. "*The Performance Analysis of Honeypot Based Intrusion Detection System for Wireless Network",* , **International Journal of Wireless Information Networks** 24(1) 2017, DOI: 10.1007/s10776-016-0330-3

Alshahrani A. *"Predication Attacks Based on Intelligent Honeypot Technique"* **International Journal of Information Science**, Vol. 12, Issue 3, Article 46 2023, Pp 1631 – 1635, https://digitalcommons.aaru.edu.jo/isl/vol12/iss3/46

Alaa B. & Safa A. "Securing a Web-Based Hospital Management System Using a Combination of AES and HMAC" **Iraqi Journal for Electrical and Electronic Engineering** 2023**,** DOI: 10.37917/ijeee.19.1.12

Anand G, Neha T & Nayankumar H. "*An Overview Of Honeypot Systems***, International Journal Of Computer Sciences And Engineering,** 7(2): 2019 394-397, Doi: 10.26438/Ijcse/V7i2.394397

Amal R. & Venkadesh P. **"***Review of Cyber Attack Detection: Honeypot System*"**. Webology,** Volume 19, Number 1, 2022 ISSN: 1735-188X DOI: 10.14704/WEB/V19I1/WEB19370

Amal R & Venkadesh P. "H-DOCTOR: *Honeypot Based Firewall Tuning for Attack Prevention***",** Measurement: Sensors, Vol, 25, 2023, https://doi.org/10.1016/j.measen.2022.100664.(**https://www.sciencedirect.com/science/article/pii/S2665917422002987)**

Abhishek S. "*Honeypots in Network Security*" **International Journal of Technical Research and Applications** Volume 1, Issue 5 2013, PP. 07-12 e-ISSN: 2320-8163, www.ijtra.com

Abiodun E Omolara J & Oludare I. "*A Comprehensive Review of Honey Encryption Scheme"*. **Indonesian Journal of Electrical Engineering and Computer Science**, Vol. 13, No. 2, 2019, pp. 649~656 ISSN: 2502-4752, DOI: 10.11591/ijeecs. v13.i2.

Akshat D, Anchit B, Nihal A & Sumithra D. *"HONEYPOT: Intrusion Detection System"* **International Journal of Education Science Technology and Engineering** 3(1): 2020 13-18, DOI: 10.36079/lamintang.ijeste-0301.66

Akshay M & Shaveta B. **"***Spark-Based Network Security Honeypot System: Detailed Performance Analysis***" International Journal of Safety and Security Engineering** 12(6): 2022 pp 737-743 DOI: 10.18280/ijsse.120610

Ala M, Ibrahim O, Ashraf A, Shadi A, Fatima Q, Dena A, Aseel A & Laith A. "*Simulation and Analysis Performance of Ad-Hoc Routing Protocols Under DDOS Attack and Proposed Solution*". **International Journal of Data and Network Science** Vol. 7 Issue 2 2023 pp. 757-764, 3ISSN 2561-8156 (Online) - ISSN 2561-8148 (Print) DOI: 10.5267/j.ijdns.2023.2.002

Amal M & Venkadesh P "Hybrid H-DOC: A bait for analyzing cyber attacker behavior". **International Journal of Electrical and Computer Engineering Systems,** Vol. 14 No. 1, 2023. https://doi.org/10.32985/ijeces.14.1.5

Amal R & Venkadesh P. *"R2NET: Storage and Analysis of Attack Behavior Patterns*" **KSII Transactions on Internet and Information Systems** Vol. 17, No. 2, 2023 295-311. DOI: 10.3837/tiis.2023.02.001.

Amit M. & Sanjay K. "*A Survey on Question Answering Systems with Classification*". **Journal of King Saud University-Computer and Information Sciences** 28.3 2016, pp. 345–361, https://doi.org/10.1016/j.jksuci.2014.10.007

Anupama M & Ammar A. *"Malware Detection Techniques: A Comprehensive Study",* **An International Interdisciplinary Journal**, Vol. 01, No. 01, 2023 pp 1 – 5

Anand G, Neha T & Nayankumar H . *"An Overview Of Honeypot Systems"*, **International Journal Of Computer Sciences And Engineering** 7(2): 2019 394-397, Doi: 10.26438/Ijcse/V7i2.394397

Ateek M, Shubhangi M. *"Data Security using Honeypot System"*, **International Research Journal of Engineering and Technology (IRJET)** Volume: 05 Issue: 03 2018 www.irjet.net p-ISSN: 2395-007 e-ISSN: 2395-0056

Barbulescu, R. & Duquesne, S. "*Updating Key Size Estimations for Pairings*". **Journal of Cryptology**. 32 2019, 1– 39. https://doi.org/10.1007/s00145-018-9280-5.

Bilal A, Dana A, Malak A, Seyed M, Tomayess I, Ibrahim A, Amin B & Sulaiman A. "*An Intelligent System for Multi-Topic Social Spam Detection in Microblogging*". **Journal Of Information Science**, 2022, https://doi.org/10.1177/01655515221124062

Bringer M., Chelmecki C & Fujinoki. H. "*A Survey: Recent Advances and Future Trends in Honeypot Research*," **International Journal of Computer Network and Information Security**, vol. 4, no. 10 2012 pp. 63, DOI: 10.5815/ijcnis.2012.10.07MECS (http://www.mecs-press.org/)

Brooke L & Weizhi M. *"A survey of deep learning-based intrusion detection in automotive applications"* **Expert Systems with Applications**". Vol. 221, 2023, 119771, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2023.119771. **(https://www.sciencedirect.com/science/article/pii/S0957417423002725)**

Cheng-H. Chou, Chi-C. Wu, Kuan-C. Lu, I-Hsien L & Jung-S. Li. *"Modbus Packet Analysis and Attack Mode for SCADA System*" **Journal of ICT, Design, Engineering and Technological Science**, vol. 2(2), 2018 pp 30-35. DOI: 10.33150/JITDETS-2.2.1

Chie H. "*Using The Modified Diffie-Hellman Problem to Enhance Client Computational Performance in A Three-Party Authenticated Key Agreement*". **Arabian Journal for Science and Engineering**. 43 (2), 2018 637–644. https://doi.org/10.1007/s13369-017-2725-6.

Christin A, Giselle C, Wesam A, Abu A & Maha S. "*A Comprehensive Survey for IoT Security Datasets Taxonomy, Classification and Machine Learning Mechanisms.*

Computers & Security, 2023, 103283, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2023.103283. **(https://www.sciencedirect.com/science/article/pii/S0167404823001931)**

Danny V. & Glen D. *"A Review of The Current State of Honeynet Architectures and Tools"*, **International Journal of Security and Networks** Vol.12 No.4, 2017, DOI: 10.1504/IJSN.2017.10009165

Danny V & Glen R. "*Ontology for Data Integration in Honeynet*". **Journal of Positive School Psychology (JPSP)** Vol. 6 No. 8 2022 ISSN: 2717-7564 (https://hjrs.hec.gov.pk/index.php?r=site%2Fresult&id=993342#journal_result)

Dansana, J., Kabat R. & Pattnaik, K. *"A Novel Optimized Perturbation-Based Machine Learning for Preserving Privacy in Medical Data.* **Wireless Personal Communication** 2023. https://doi.org/10.1007/s11277-023-10363-x

David P. "*Just & Unjust War, Uses of Force & Coercion: An Ethical Inquiry with Cyber Illustrations*", Daedalus, vol. 145, no. 4, 2016, pp. 37–49. **JSTOR, http://www.jstor.org/stable/24916782.**

Emilio I. "*What is the Role of Cyber Operations in Information Warfare*"? **Journal of Strategic Security,** vol. 14, No 4, 2021 Private Sector, iasiello@aol.com

Eiichiro F. "*All-But-Many Encryption*", **Journal of Cryptology,** 31**,** 2018 pp 226–275, DOI https://doi.org/10.1007/s00145-017-9256-x

Farhan S & Shamik S. "*Modeling and analyzing Attacker Behavior in IoT Botnet Using Temporal Convolution Network (TCN)*", Computers & Security, Volume 117, 2022, 102714, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2022.102714. **(https://www.sciencedirect.com/science/article/pii/S0167404822001092)**

Fransiska S. Mukti & Muhammad R. Sukmawan. *"Integration of Low Interaction Honeypot and ELK Stack as Attack Detection Systems on Servers"* **Jurnal Penelitian Pos dan informatika** 11(1), 2021, DOI: 10.17933/jppi.v11i1.336.

Francesco S, Anastasia M, Daniyar G, Philip T, Ross B & Nicholas A. "*Unsupervised Attack Pattern Detection in Honeypot Data Using Bayesian Topic Modelling*". **Journal of Cryptography and Security,** 2023, https://doi.org/10.48550/arXiv.2301.02505

Gbenga I & Yasser M. *"Systematic Review of Graphical Visual Methods in Honeypot Attack Data Analysis* ", **Journal of Information Security**, 13, 2022 pp 210-243 https://www.scirp.org/journal/jis ISSN Online: 2153-1242 ISSN Print: 2153-1234

Gupta B & Gupta A. "*Assessment of Honeypots: Issues, Challenges and Future Directions*". **International Journal of Cloud Applications and Computing (IJCAC)** 8(1) 2018 |Pp 34 DOI: 10.4018/IJCAC.2018010102

Gururaj L., Swathi H., Trupti, R. "*Analysis of Preventive Measures Against DDoS Attacks in Smart Grid*. **Journal of the Institution of Engineers** 104, 2023 pp297–303 https://doi.org/10.1007/s40031-022-00844-1

Hafsat M, Longe .O & Baale A.. "*Towards the Development of a Machine Learning Enhanceed Framework for Honeypot and CAPTCHA Intrusion Detection Systems*" **Advances in Multidisciplinary and Scientific Research Journal Publication** 2022 DOI: 10.22624/AIMS/ACCRABESPOKE2022/V34P4

Hangfeng H, Yi C, Wenhai Q, Maoli W & Xiaoming C. "*Observer-based Resilient Control of Positive Systems with Heterogeneous Dos Attacks: A Markov Model Approach*", **Journal of the Franklin Institute,** Volume 359, Issue 1, 2022, Pages 272-293, ISSN 0016-0032, https://doi.org/10.1016/j.jfranklin.2021.04.034. (https://www.sciencedirect.com/science/article/pii/S0016003221002519)

Ihsan H. Abdulqadder A, Deqing Z & Israa T. "*The DAG Blockchain: A Secure Edge Assisted Honeypot for Attack Detection and Multi-Controller Based Load Balancing in SDN 5G*". **Future Generation Computer Systems** Volume 141, 2023, Pages 339-354

Ismael S, Al-Ta'i T, Hussain A, & Sabaa J. "*Online Intrusion Detection System Using C4.5 Algorithm with Honeypot"*, **Journal of Engineering and Applied Sciences** 15(5): 2019 1127-1132 DOI: 10.36478/jeasci.2020.1127.1132 2019

Irini L, Shreyas S, Emmanouil V & Dimitris G. "*A Decentralized Honeypot for IoT Protocols based on Android devices"*, **International Journal of Information Security** 21(1), 2022, DOI: 10.1007/s10207-022-00605-7

Jianxun T, Mingsong C, Haoyu C & Yu H. *"A New Dynamic Security Defense System Based On TCP_REPAIR And Deep Learning"*, **Journal of Cloud Computing** 12(1), 2023, DOI: 10.1186/s13677-022-00379-2, License

Kai C, Zhan W, Dongkun L & Mu R. *"The TaintDroid Based Honeypot Monitoring System for Embedded Device"*, **Journal of Physics Conference Series** 2203(1):2022 012077, DOI: 10.1088/1742-6596/2203/1/012077.

Kang M & Kang J. "*Intrusion Detection System Using Deep Neural Network for In-Vehicle Network Security*". **PLoS ONE Journa**l 11(6) 2016 e0155781. https://doi.org/10.1371/journal.pone.0155781

Kathan P. & Dhaval M. "*National Security Threats in Cyberspace*", National **Journal of Cyber Security Law** Volume 4, Issue 1 2021 pp 109–114, DOI (Journal): 10.37591/NJCSL http://lawjournals.celnet.in/index.php/njcsl/index,

Kuan-C L., I-Hsien L, Jia-W. Liao & Chu-F. Li. *"Evaluation and Build to honeypot System about SCADA Security for Large-Scale IoT Devices"*, **Journal of Robotics Networking and Artificial Life** 6(3), 2019, DOI: 10.2991/jrnal.k.191202.008, License CC BY-NC

Kyungroul L, Jaehyuk L & Kangbin Y. "*Classification and Analysis of Malicious Code Detection Techniques Based on the APT Attack*". **Journal of Applied Sciences,** Volume 13 Issue 2023 5 10.3390/app13052894

Lidong W, Reed M, Patti D & Terril F. "Predictive Modelling of a Honeypot System Based on a Markov Decision Process and a Partially Observable Markov Decision Process". **Journal of Applied Cybersecurity & Internet Governance** 2023, ISSN: 2956-3119 | E-ISSN: 2956-4395, DOI: 10.5604/01.3001.0016.2027

Logeshwaran J, Ramesh G & Aravindarajan V. "*A Secured Database Monitoring Method to Improve Data Backup and Recovery Operations in Cloud Computing*". **BOHR International Journal of Computer Science** Vol. 2 No. 1 2023: https://journals.bohrpub.com/index.php

Maryam M & Jaber k. "Using Rootkits Hiding Techniques to Conceal Honeypot Functionality" **Journal of Network and Computer Applications** Volume 214, 2023, 103606, https://doi.org/10.1016/j.jnca.2023.103606

Matthew L., Christopher C, & Hiroshi F. "*A Survey: Recent Advances and Future Trends in Honeypot Research*". **International Journal of Computer Network and Information Security,** 4.2018 doi: 10.5815/ijcnis.2012.10.07.

Mohd A, Muhammad A, Mohamed Y, Haryani K, Aditya M, Yohan P & Chrisando R. "*Deployment of Honeypot and SIEM Tools for Cyber Security Education Model in UITM*". **International Journal of Emerging Technologies in Learning**. 2022, Vol. 17 Issue 20, p149-172. 24p.

Morteza S, Christelle N, Kurt F & Elias B. "*A Comprehensive Survey of Recent Internet Measurement Techniques for Cyber Security*", **Computers & Security,** Vol. 128, 2023, 103123, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2023.103123. **(https://www.sciencedirect.com/science/article/pii/S0167404823000330)**

Movva, S, Nikudiya S, Basanaik S, Damodar R & Hanumanthu B. "*Intelligent IDS: Venus Fly-Trap Optimization with Honeypot Approach for Intrusion Detection and Prevention*". **Journal of Wireless Personal Communication** 128, 2023 1041–1063. https://doi.org/10.1007/s11277-022-09988-1

Mphago, B. & Mpoeleng, S. "*Deception in Web Application Honeypots: Case of Glastopf*". **International Journal of Cyber-Security and Digital Forensics**, 6(4), 2017 pp. 179-185, ISSN: 2305-0012

Mu A, Mu J & Mahadik S. "*Data Security using Honeypot System"*, **International Research Journal of Engineering and Technology (IRJET)** Vol: 05 Issue: 03 | 2018 www.irjet.net e-ISSN: 2395-0056

Muhammad A, Mohamed Y, Haron, H, Kurniawan, A & Muliono, Y. "*Deployment of Honeypot and SIEM Tools for Cyber Security Education Model In UITM"*. **International Journal of Emerging Technologies in Learning** (Online) Vol. 17, Iss. 20, 2022: 149-172. DOI:10.3991/ijet. v17i20.32901

*Naidu D, & Jha M "Detection Technique to Trace IP Behind VPN/Proxy using Machine Learning"***. International Journal of Next-Generation Computing**. Feb2023, Vol. 14 Issue 1, p216-222. 7p. EBSCO Information Services

Neha T, Nayankumar H & Anand G. "*An Overview Of Honeypot Systems***", International Journal Of Computer Sciences And Engineering** 7(2): 2019 394-397, Doi: 10.26438/Ijcse/V7i2.394397.

Nisha T & Dhanya P. *"Insider Intrusion Detection Techniques: A State-of-the-Art Review"*. **Journal of Computer Information Systems,** 2023, https://doi.org/10.1080/08874417.2023.2175337

Park S, Li G, & Hong J, "*A Study on Smart Factory-Based Ambient Intelligence Contextaware Intrusion Detection System Using Machine Learning*," **Journal of Ambient Intelligence and Humanized Computing**, 11, 2020 pp 1405–1412, https://doi.org/10.1007/s12652-018-0998-6.

Paul B & Rao M. "*Zero-Trust Model for Smart Manufacturing Industry*". **Applied Sciences Journal.** 13(1) 2023 :221. https://doi.org/10.3390/app13010221

Pooja K & Ankit K. "*A Comprehensive Study of DDoS Attacks over IoT Network and their Countermeasures*", **Computers & Security,** Volume 127, 2023, 103096, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2023.103096. **(https://www.sciencedirect.com/science/article/pii/S0167404823000068)**

Priya, V & Chakkaravarthy S. "*Containerized Cloud-Based Honeypot Deception for Tracking Attackers*". **Scientific Reports Journal** 13, 1437 2023. https://doi.org/10.1038/s41598-023-28613-0

Rajesh D, Hussan M. & Vastav B. *"Network Protection using Honeypots"*, **International Journal of Innovative Technology and Exploring Engineering** (IJITEE) ISSN: 2278-3075 (Online), Volume-9 Issue-6, 2020

Rejwan B. & Masud R. "*A Detailed Study on Web-Based-Honeypot to Propose Mitigation Framework in Web Applicatio*". **Computer Aided Engineering Journal** 2019, SSRN Electronic Journal, DOI: 10.2139/ssrn.3723098

Risa E, Muhammad W. & Prabowo A. "*Implementasi Intrusion Prevention System* (IPS) OSSEC dan Honeypot Cowrie" **Jurnal Sisfokom (Sistem Informasi dan Komputer)** 11(1): 2022 73-78, DOI: 10.32736/sisfokom.v11i1.1246, License CC BY 4.0

Roye L. "*Incorporating a Honeyfarm with Mlffnn IDS for Improving Intrusion Detection*". **International Journal of Advanced Research in Computer Science**. Vol. 14 Issue 1, 2023 pp1-4. 4.

Rusabh K. & Rajapraveen.K. *"Secured Honeypots To Understand Attacks To Control Systems",* **International Journal for Science and Advance Research in Technology (IJSART)** - Volume 5 Issue 8 2019 ISSN [ONLINE]: 2395-1052

Sakthidasan K & Kim B. *"Deep Learning Based Energy Efficient Optimal RMC-CNN Model for Secured Data Transmission and Anomaly Detection in Industrial IOT"*, **Sustainable Energy Technologies and Assessments,** VoL. 56, 2023, 102983, ISSN 2213-1388, https://doi.org/10.1016/j.seta.2022.102983. **(https://www.sciencedirect.com/science/article/pii/S2213138822010311)**

Sara B, Mauro C, Luca P & Pier P. "*Social Honeypot for Humans: Luring People through Self-Managed Instagram Pages*". **Journal of Social and Information Networks, cs.SI), Artificial Intelligence (cs.AI), Cryptography and Security (cs.CR)** https://doi.org/10.48550/arXiv.2303.17946

Sharma, P., Kapoor, S. & Sharma, R. "*Ransomware Detection, Prevention and Protection in IoT Devices Using ML Techniques Based on Dynamic Analysis Approach*". **International Journal of System Assurance Engineering and Management** 14, 2023 287–296. https://doi.org/10.1007/s13198-022-01793-0

Shah A. "*Evaluating Network Forensics Applying Advanced Tools*". **International Journal of Advanced Engineering, Management and Science**, Vol 9 No 4 2023, http://journal-repository.theshillonga.com/index.php/ijaems/article/view/6178

Sakama C., Caminada M, & Herzig A. "*A formal account of dishonesty*," **Logic Journal of IGPL,** vol. 23, no. 2, 2015 pp. 259–294, https://doi.org/10.1093/jigpal/jzu043

Santhosh K, Selvi M & Kannan A "*A Comprehensive Survey on Machine Learning-Based Intrusion Detection Systems for Secure Communication in Internet of Things*" **Journal of Computational Intelligence and Neuroscience,** 2023 Article ID 8981988 | https://doi.org/10.1155/2023/8981988

Saurabh C, Rakesh K & Sehgal, Ram S. "*Honeypot Baselining for Zero Day Attack Detection*". **International Journal of Information Security and Privacy (IJISP)** 11(3) 20 DOI: 10.4018/IJISP.201707010617,

Savvas Z, Michael S, Jeremy B, & Nicolas K. "*The Web of False Information: Rumors, Fake News, Hoaxes, Clickbait, and Various Other Shenanigans*". **Journal of Data and Information Quality** Vol. 11, No. 3 Article No.: 10 2019 pp 1–37https://doi.org/10.1145/3309699

Sathiyandrakumar S & Deepalakshmi P. "*Enhancing the Security in Cyber-World by Detecting the Botnets Using Ensemble Classification Based Machine Learning*", Measurement Sensor Volume 25, 2023, 100624, ISSN 2665-9174, https://doi.org/10.1016/j.measen.2022.100624. **(https://www.sciencedirect.com/science/article/pii/S2665917422002586)**

Shreyas S. Jens M. Pedersen & Emmanouil V. "*Gotta Catches 'Em All: A Multistage Framework for Honeypot Fingerprinting*", **ACM Journal of Digital Threats: Research and Practic**e. 2023 DOI: 10.1145/3584976,

Sheetal D. Gokhale, Ashwini D. & Irfan S. *"Industrial Control Systems Honeypot: A Formal Analysis of Conpot"*, **International Journal of Computer Network and Information Security** 12(6): 2020 44-56, DOI: 10.5815/ijcnis.2020.06.04

Sikos, L., Valli, C., Grojek, A. "*Camdec: Advancing Axis P1435-LE Video Camera Security Using Honeypot-Based Deception"*. **Journal of Computer Virology and Hacking Technque** 2023. https://doi.org/10.1007/s11416-023-00463-4

Smita k. Jawale, Rishi M, Vivek M & Niyoshi M *"Intrusion Detection System using Virtual Honeypots",* **International Journal of Engineering Research and Applications,** 2022**,** SSN: 2248-9622

Solomon Z. & Avadhani S. *"Honeypot System for Attacks on SSH Protocol",* **International Journal of Computer Network and Information Security Septembe**r 8(9): 2016 19-26 DOI: 10.5815/ijcnis.2016.09.03

Stefan M. *"Honeypot Implementation in a Cloud Environment*", **Journal of Cryptography and Security**, 2023, https://doi.org/10.48550/arXiv.2301.0071

Sviatoslav V, Vitalii S, Ivan O, Yevhenii K & Ivan T. "*A Model of Decoy System Based on Dynamic Attributes for Cybercrime Investigation*". **Eastern-European Journal of Enterprise Technologies,** 1(9 (121), 2023 6–20. doi.10.15587/1729-4061.2023.273363

Tang, J., Chen, M. & Chen, H. "*A New Dynamic Security Defense System Based on TCP_REPAIR and Deep Learning*". **Journal of Cloud Computing** 12, 21 2023. https://doi.org/10.1186/s13677-022-00379-2

Veronica V, Maria R & Sebastian G. *"Attacker Profiling Through Analysis of Attack Patterns in Geographically Distributed Honeypots".* **Journal of Cryptography and Security (cs.CR); Networking and Internet Architecture (cs.NI),** https://doi.org/10.48550/arXiv.2305.01346

Vincent N, Mohamed K, Eric K & Matthieu H. "*Setup and Deployment of a High-Interaction Honeypot: Experiment and Lessons Learned*" **Journal in Computer Virology,** 7(2):2011, 143–157. doi: 10.1007/ s11416-010-0144-20

Wael A. & Alrashdan T. "The Effect of Using Honeypot Network on System Security "**International Journal of Data and Network Science** 6:2022 pp 1 – 6, DOI: 10.5267/j.ijdns.2022.5.010

Waqas A., Muhammad A., Sabreena N & Farhana W. "*Detection and Analysis of Active Attacks using Honeypot*". **International Journal of Computer Applications** (0975 – 8887) Volume 184 – No. 50, 2023 IJCATM: www.ijcaonline.org

Wenjun F, Zhihui D, David F & Victor V. "*Enabling an Anatomic View to Investigate Honeypot Systems: A Survey*" **EEE Systems Journal,** vol. 12, no. 4, pp. 2018 3906-3919, doi: 10.1109/JSYST.2017.2762161.

Xingyuan Y, Jie Y, Hao Y, Ya K, Hao Z & Jinyu Z. "*A Highly Interactive Honeypot-Based Approach to Network Threat Management*", **Journal of Future Internet,** 2023, 15(4), 127; https://doi.org/10.3390/fi15040127

Xingsheng Q, Frank J, Mingcan C & Robin D *"Hybrid cyber defense strategies using Honey-X: A survey*". Computer Networks, Volume 230, 2023, 109776, ISSN 1389-1286, https://doi.org/10.1016/j.comnet.2023.109776 **(https://www.sciencedirect.com/science/article/pii/S1389128623002219)**

Yanling Z., Jing C. & Yunxiang Z. "*Research on Network Security Situation Awareness Based on Netlogo*". **Research Square**, 2023, DOI: https://doi.org/10.21203/rs.3.rs-2413748/v1

Yaser A. "*Improving Intrusion Detection Systems Using Artificial Neural Networks*". **ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal**, Vol. 7 No. 1 2018 https://doi.org/10.14201/ADCAIJ2018714965

Yuan H, Changyou X, Ke D, Guomin Z & Lihua S. "*A Differential Privacy Based Multi-Stage Network Fingerprinting Deception Game Method*", **Journal of Information Security and Applications**, Volume 74, 2023, 103460, ISSN 2214-2126, https://doi.org/10.1016/j.jisa.2023.103460.(https://www.sciencedirect.com/science/article/pii/S2214212623000443)

Yuhui Z, Zhenxiang C, Qiben Y, Shanshan W, Enlong L, Lizhi P & Chuan Z. "*Mining Function Homology of Bot Loaders from Honeypot Logs*". **Journal of Cryptography and Security**, 2022, arXiv:2206.00385 https://doi.org/10.48550/arXiv.2206.00385

Yung-She L & Chin-Feng L. "*Ransomware Detection and Prevention through Strategically Hidden Decoy File*". **International Journal of Network Security**, Vol.25, No.2, 2023 PP.212-220, (DOI: 10.6633/IJNS.202303 25(2).04)

Zhang D, Gang F., Yang S., & Dipti S. "*Physical Safety and Cyber Security Analysis of Multi-Agent Systems: A Survey of Recent Advances*." **IEEE/CAA Journal of Automatica** Sinica 8, no. 2, 2021: 319-333

Magazine Articles

Abbasgholi P, Mohammad E. Akbari, Mina Z. Lighvan & Asghar C. "*Machine Learning-Based Early Intrusion Detection System in Industrial LAN Networks Using Honeypots"* 2021, DOI: 10.21203/rs.3.rs-1122586/v1.

David C. Stuart. **"**The Case for Next-Gen Intrusion Prevention to Protect Digital Business**",** Cisco SecurityNotice 2016

Fraunholz, D., Zimmermann, M., & Schotten, H. "*An Adaptive Honeypot Configuration, Deployment and Maintenance Strategy*". 19th International Conference on Advanced Communication Technology (ICACT) 2017 pp. 53-57. DOI: 10.23919/ICACT.2017.7890056

Hoffpauir, K., Markle, N., Meadows, C., & Pittman, J. "*A Taxonomy for Dynamic Honeypot Measures of Effectiveness*". Cryptography and Security (cs.CR), 2020, https://doi.org/10.48550/arXiv.2005.12969

Huang, L., & Zhu, Q. "*Adaptive Honeypot Engagement Through Reinforcement Learning of Semi-Markov Decision Processes*". International Conference on Decision and Game Theory for Security 2019 pp. 196-216. DOI: 10.1007/978-3-030-32430-8_13

Nadiya E. Kamel, Mohamed E, Youssef L & Raja T. *"A Smart Agent Design for Cyber Security Based on Honeypot and Machine Learning*", Security and Communication Networks 2020, https://doi.org/10.1155/2020/8865474

Vaishali S. & Madhav M. *"Efficacy Measuring Framework for the Assessment of Dynamic Honeypot*". 2021 International Conference on Advances in Computing, Communication, and Control (ICAC3), 2021 DOI: 10.1109/ICAC353642.2021.9697296

Vitaly V. & Sergei A. "*Architecture of the Honeypot System for Studying Targeted Attacks*", XIV International Scientific-Technical Conference on Actual Problems of Electronics Instrument Engineering (APEIE), 2018, pp. 202-205, doi: 10.1109/APEIE.2018.8545323.

Electronic Source

Efrén M, Carlos R & Adam D. "*HoneyPLC: A Next-Generation Honeypot for Industrial Control Systems*", ACM SIGSAC Conference on Computer and Communications Security. 2020 pp 279–291https://doi.org/10.1145/3372297.3423356

Lockheed M. "*Cyber Kill Chain: Seven Steps of a Cyberattack*". 2022 https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html

Thangarasu, N. & Selvakumar A. "*Improved Elliptical Curve Cryptography and Abelian Group Theory to Resolve Linear System Problems in Sensor-Cloud Cluster Computing". Cluster* Computer. 22. 2019 pp 13185–13194 https://doi.org/10.1007/s10586-017-1573-1.

Kai C, Zhan W, Chengcheng Z & Haohua M. *"The Research on Network Function Virtualization Based Network Honeypot"*, Proceedings of the 12th International Conference on Computer Engineering and Networks, 2022, pp 1481–1487 DOI: 10.1007/978-981-19-6901-0_156

Meatasit K, Hiroshi E & Hideya O. **"***SDNHive: A Proof-of-Concept SDN and Honeypot System for Defending Against Internal Threats***",** Conference: 11th International Conference on Communication and Network Security, 2021, Pp 9–20 DOI: 10.1145/3507509.3507511

List of Journal Publications

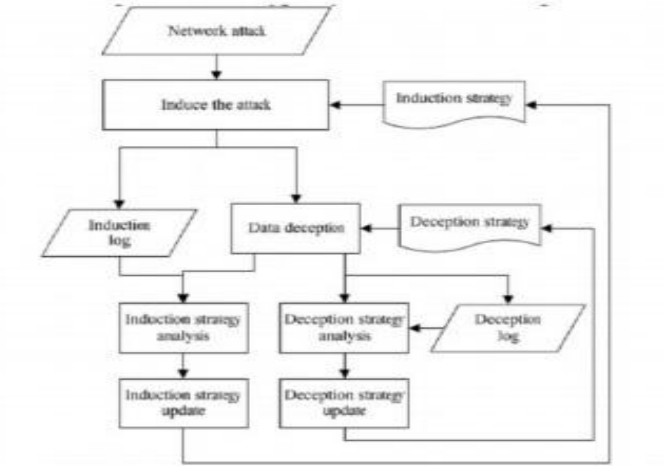
Osang F, Umoren I & Owolabi A. "*Implementing an Enhanced Procurement Management System Using Decision Support Techniques*", **Journal of Computer Science and Its Application** 28(1), September 2021, DOI: 10.4314/jcsia.v28i1.5

Owolabi A. "*Discovery of ICT for the Growth and Development in Sub Sahara Africa: Utilization, Enlightenment, Efforts and Challenges*", **International Journal of Informatics and Communication Technology** (IJ-ICT), Institute of Advanced Engineering and Science, 2012, ISSN: 2252-8776-IJ- ICT Vol 1 No 1, July 2012: 26

Popoola O, Wasiu J & Owolabi A. "*Determination of Traffic Delay at Selected Intersection within Ilorin Metropolis*", **American Journal of Engineering Research (AJER) American Journal of Engineering Research** (AJER) e-ISSN: 2320-0847 p-ISSN: 2320-0936 Volume-4, 2015, Issue-9, pp-176-180 www.ajer.org

Appendices

Appendix I: *Basic Processing Flow of Honeypot System*



Appendix II: Comparison Study of Different Methods

Authors	Title	Method	Outcome
(Yicheng et. Al, 2019)	In an IoT-based cloud computing environment, secure authentication and key agreement scheme is presented	AKAP	Low accuracy
(Poorvika et.al, 2020)	Detecting and preventing cloud intruders by using a honeypot network	Honeypot Protocol	Reduce attack rate
(Feifei Wang et. Al, 2019)	Authentication Protocol Based on ECC with High Security and Efficient Performance	ECC based AAP	Low throughput
(Yang et.al, 2019)	Security and privacy in VANETs with a certificate-less conditional authentication scheme	CPPAP (Conditional Privacy-Preserving Authentication Protocol)	Reduce communication cost

Appendix III: Different Methods and Their Features

Paper	Method detail	Mitigated attacks	Vulnerabilities/Limitation
(Kapczynski and Lawnik, 2019)	The use of variable key length cyphers	This system is designed to resist various attacks, such as side-channel attacks, related-key attacks, and plain text attacks	Execution time and space are vastly increased.
(Aggarwal and Maurer, 2016)	RSA factoring using the generic ring algorithm	RSA mitigation of factoring issues	Various cryptoanalysis attacks are possible.
(Hwang et al., 2016)	Using pairless cryptography to encrypt certificates	Protects against attacks using chosen cypher texts	Due to its reliance on bandwidth, the architecture is prone to Denial-of-Service attacks.
(Fujisaki, 2018)	Based on a binary string, this method involves public-key encryption with apt length.	A defence against man-in-the-middle attacks.	Attacks that interfere with service may result in denial of service.
(Hazay et al., 2018)	Utilize two-party distributed factoring to resolve the factoring problem.	Defend yourself against malicious attacks.	The size of the application and the execution time increases dramatically.
(Dwivedi, 2011)	Distributed-transforming encoders for message recovery.	Ensure security against brute force attacks.	This vulnerability can be exploited by known plain text attacks.
(Biswas and Mohit, 2016)	Implementing RSA within DES	Protection against a variety of threats.	An attack using known-cypher text is possible, as well as brute force attacks
(Chie, 2018)	Generating session keys based on key agreement schemes.	Passive and active defence models.	Attacks by third parties are possible.
(Barbulescu and Duquesne, 2017)	Using a variant of NFS, suggest a novel key size.	Reduce the risk of DOS, impersonation and replay attacks.	It cannot be accessed by a multi-server environment.
(Thangarasu and Selvakumar, 2018)	Modifying the ECC algorithm to secure session keys.	Intruder attacks can be mitigated.	Attacked by traditional methods.

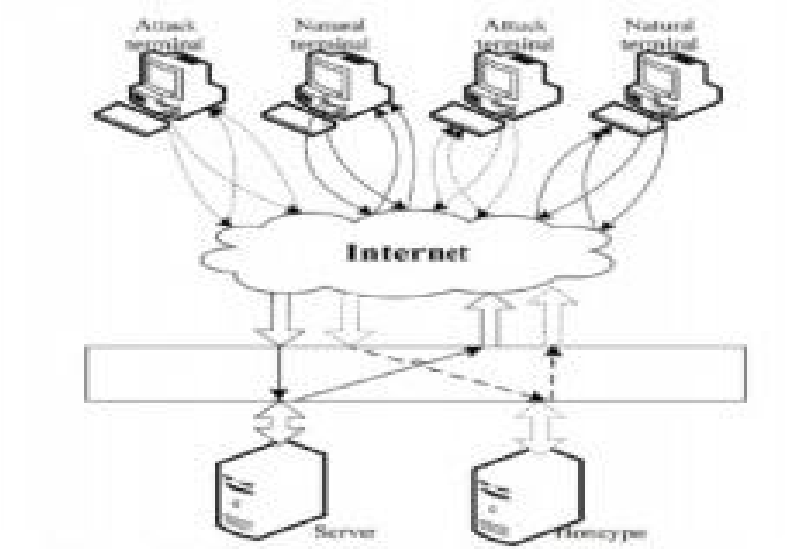
Appendix IV*:* Comparative Study of Different Techniques

Algorithm / Technique Used	Reference Paper	Purpose of IDS	Advantage	Limitation/ Future Scope
KDD and clustering	(Chen et al. 2017)	Detecting novel anomalies referred to as NEC	It is not necessary to have quality labelled datasets.	A large number of false positives and a high number of false positives.
Decision tree, random forest, K-NN	(Anbar et al. 2016)	To accurately detect potential attack	Produce impressive and efficient results in detecting IPV4-based attacks.	IPV6 attack cannot be detected yet.
GPFISClass (genetic programming fuzzy inference system for classification)	(Ahmim, A., & Zine, N.G. 2015)	Solving the classification problem in IDS.	Higher classification accuracy.	Introducing the new hybrid GFS that combines neural networks with GFS.
Epigenetic algorithm	(Ghazi et al. 2016)	The future offspring of this couple.	By preventing diseases based on environmental factors that are not related to the sequenced genes, it helps prevent more preciously the curable.	A shorter time is spent to obtain optimal solutions when there are fewer iterations.

Appendix V*:* Comparative Study of Different Approaches

Ref.	Approaches	Strengths	Weaknesses
(Choi et al, 2018)	Botnet, IoT botnet	Web service is available for easy monitoring of IoT device health and is useful for smart factories with many IoT devices.	Limited capacity
(Park et al., 2018)	Smart factory detection using machine learning.	Cost reduction	Low detection rate, high complexity and uncertainty.
(Wang et al.,2020)	Exploiting IoT honeypots for detection.	In addition, since only parts of the system are implemented, the gathering of information is rapid.	Stacks of unneeded data accumulate.
(Wang et al., 2020)	Botnet detection using Machine learning.	In this case, the combination of a flow-based and a graph-based analysis achieves a detection accuracy of 99.94%, exceeding individual detectors at each stage.	To ensure security, it is necessary to randomly specify the size of the packets and the number of packets in each flow so that they are not detected. Flow-based detectors cannot be quickly applied.
(Vishwakarma, 2019)	Machine learning detection with the Honey Pot	With the help of machine learning, it has developed a solution to detect botnets using honeypots. Honeypots log newly released malware functions so that they can be identified in the future.	The functions of the system vary depending on how well it performs.

Appendix VI: *Honeypot Working Principles*



Appendix VII: *Different Between Low Interaction and High Interaction Honeypot*

High Interaction Honeypot	Low Interaction Honeypot
No imitation, legitimate OS, and facility are provided.	Operating systems and services are followed by a solution.
Can be difficult to install or use.	Installation and deployment are uncomplicated.
Large information can be caught.	Apprehend limited amount of data.
Huge risk because for interaction real OS is provided.	Low risk due to imitated Services

Appendix VIII: *Low, Medium and High Interaction Honeypot*

	Low interaction honeypot	Medium interaction honeypot	High interaction honeypot
Installation and configuration	Easy	Involved	Difficult
Redesign and development	Easy	Involved	Difficult
Maintenance	Easy	Involved	Difficolt

Information gathering	limited	Variable	extensive
Level of risk	Low	Medium	High
Resource utilization	Low	Medium	High
Compromised wished	No	No	Yes
Knowledge to develop	Low	High	Medium high
Knowledge to run	Low	Low	High
Implementation cost (hardware,licenses.... etc)	Low	Low medium	Medium high
Real OS	No	No	Yes

Appendix IX: Comparison of Various Honeypot Tools

	Man-Trap	BOF	DTK	Spec-ter	Honeyd	HIHAT
Interaction level	High	Low	Mid-dle	High	Low	High
Freely available	No	No	Yes	No	Yes	Yes
Log file support	Yes	No	Yes	Yes	Yes	Yes
OS simulation	Yes	No	Yes	Yes	Yes	Yes
Supported services	Unres-tricted	7	Limited	13	Unres-tricted	Unrestricted
Support Graphical user interface	Yes	Yes	No	Yes	Yes	Yes

Appendix X: *Comparison of Different Mobile Honeypots*

Honeypot	Interaction Level	Armature	Strength	Weakness
Honeypot Labsac (Oliveira et al., 2013)	Low	Yes	Active Monitoring and Software reuse	Less alluring to Attackers
Honeydroid (Mulliner, Liebergald, Lange, 2011)	High	No	Seize attacks commencing from web applications, mobile network, and Applications	Do not acquit like Android OS and a case of replay attacks.
Nomadic Honeypot (Liebergeld, Lange, & Mulliner, 2013)	Low	No	Cannot be accustomed to nurturing attacks.	Only useful when actively used by the user.
Hostage (Vasilomanolakis et al., 2013)	Low	Yes	Can be used by the prosaic user.	Battery drainage and challenge of rooted Android Device.

Bio-data

Personal Data

Name:	Abimbola Basiru Owolabi
Sex:	Male
Marital Status	Married
Date of Birth:	20TH September. 1980
Place of Birth:	Ikosu Ekiti
Home Town:	Ikosu Ekiti
Local Govt. Area:	Moba Local Govt.
State of Origin:	Ekiti State
Nationality:	Nigerian
No of Children:	Four
Next of kin:	Owolabi Folasade C.
Religion:	Christianity
Tel:	**+234 7038457455, *+234 8084662112***
E-mail:	**aowolabi@noun.edu.ng**
	abimcomputers75@gmail.com
Years of Professional Experience in ICT:	Seventeen (17) Years
Years of Teaching Experience:	Twelve (12) years
Present Position:	Head, ICT Unit, NOUN Ibadan Study Centre
Present Level:	Chief System Analyst

Institution Attended with Dates

1:	University of Ibadan, Nigeria	2009
2:	National Open University of Nigeria, Lagos	2007
3:	Federal Polytechnic Offa, Nigeria	2003
4:	Federal Polytechnic Offa, Nigeria	1996-1998
5:	Efon High School, Efon Alaaye, Ekiti State, Nigeria	2001
6:	Lower Holy Pry. School, Aisegba Ekiti, Nigeria	1983-1988

Qualification Obtained with Dates

1	Master in Computers Science (MSc)	2011
2	Post Graduate Diploma in Info. Tech. (PGD)	2009
3:	National Youth Service Corps (NYSC)	2005
4:	Higher National Diploma in Computer Science (HND)	2003
5:	National Diploma in Computer Science (ND)	1998
6:	Secondary School Cert. (SSCE)	1994

Conferences Paper Presentation

Owolabi A. *Opportunity Knowledge Transfer Through Open and Distance Learning in Sub-Sahara Africa*. 2014 The European Conference Technology in the classroom Brighton, United Kingdom

Owolabi A. *Bro: An Open-Source Network Intrusion Detection System*, 2014 ISIS-Key West International Multidisciplinary Academic Conference, Miami USA, 2013.

Owolabi A. *Institutional Reform and Change Management: Managing Change in Public Sector Organization*, Global Academic Network International Conference, Ottawa Canada, 2013.

Owolabi A. *Design and Simulation of Electronic Collaboration (e Collaboration) using Mobile Phone for Mobile Communication*, International Conference on ICT for Development in Africa 2011 in Otta Nigeria, 2010.

Owolabi A. *An Annotation Model for Evaluating Learning Level Coupled with Learning Type in A Multimedia Environment*. International Educational Technology Conference (IETC) 2011 Istanbul, Turkey, 2011.

Owolabi A. *Discovery of ICT for the Growth and Development in Sub Sahara Africa: Utilization, Efforts, and Challenges.* Africa Council for Distance Education (ACDE) 2011 at Open University of Tanzania, Dar es Salam, 2011.

Owolabi A. *Employee's Conformity to Information Security Policies in Nigerian Business Organization - A Factor Effect Study,* (Published by Computing and Information Systems &Development Informatics Journal (CISDI/Submission/ Vol 3 No 2 2012 TGT Paper 24).

Interest Reading, Traveling, and Researching

Referees

Prof. Olumide Babatope Longe
Academic City University College,
Accra,
Ghana
+233 59 547 9930

Prof. Seyi Osunade
Computer Science Dept.
University of Ibadan
Nigeria
+234 803 326 4588

Dr. Madu Galadima
National Open University of Nigeria
Jabi, Abuja
FCT, Nigeria
+234 806 003 3036

10 – 02 - 2023

---------- ----------

Signature Date

The University Compliance Certification

This is to certify that this thesis by **Abimbola Basiru OWOLABI** with Matriculation Number **LCU/PG/002388** in the Department of Computer Science, Faculty of Natural and Applied Sciences, Lead City University, Ibadan is in full compliance with the approval of the University's format and style.

---------- ----------
Signature Date

OWOLABI BASIRU ABIMBOLA PHD LCU LIBRARY

ORIGINALITY REPORT

15%	14%	4%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	discovery.ucl.ac.uk Internet Source	4%
2	www.webology.org Internet Source	3%
3	pdfcoffee.com Internet Source	2%
4	www.iraj.in Internet Source	2%
5	repository.stcloudstate.edu Internet Source	1%
6	content.sciendo.com Internet Source	1%
7	B. B. Gupta, Alisha Gupta. "chapter 68 Assessment of Honeypots", IGI Global, 2020 Publication	1%
8	www.packtpub.com Internet Source	1%
9	www.mecs-press.org Internet Source	1%