Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X Volume 3, Issue 5, 2025

# DESIGNING AN ADAPTIVE HONEYPOT FOR ADVANCED CYBERSECURITY THREAT DETECTION

**Sidra Tul Muntaha[*1], Fareeha Ashraf[2], Iqra Shahzad[3], Jawaid Iqbal[4]**

[*1]*Master in Software Engineering, Riphah International University, Islamabad*
[2]*Master in Computer Science, Capital University of Science and Technology, Islamabad*
[3]*Master in Software Engineering, Riphah International University, Islamabad*
[4]*Assistant Professor, Faculty of Computing, Riphah International University, Islamabad*

[*1]muntasidra@gmail.com , [2]fareeha.ashraf01@gmail.com , [3]iqra22428@gmail.com, [4]jawaid.iqbal@riphah.edu.pk

DOI: https://doi.org/10.5281/zenodo.15542734

**Keywords**
*Honeypot, Low Interaction, High Interaction, Deception methods, AI/ML Integration, Modern deployment approaches*

**Article History**
*Received on 21 April 2025*
*Accepted on 21 May 2025*
*Published on 29 May 2025*


**Corresponding Author: ***
**Sidra Tul Muntaha**

**Abstract**
*Honeypots, which mimic real systems and attract attackers into controlled environments, are essential parts of contemporary cybersecurity. They are made to identify, evaluate, and lessen cyberthreats. The categories of honeypots, such as low-, medium-, and high-interaction kinds, as well as hybrid models that maximize resource use and threat intelligence collection, are thoroughly examined in this research. The efficiency of tools like Honeyd, Dionaea, Kippo, and Honeypot-as-a-Service (HaaS) in cloud, industrial, and Internet of Things ecosystems is evaluated. Honeypots are incorporated with new tactics, such as quantum-enhanced unpredictability and Deepfake-driven deception, to increase their resistance to APTs and zero-day vulnerabilities. The integration of ChatGPT to dynamically engage attackers and collect actionable intelligence, as well as containerized honeypot deployments utilizing Kubernetes, are also highlighted in the paper. Their usefulness is illustrated by real-world use cases, such as ransomware detection and web portal honeypots. To handle changing cybersecurity issues, this study suggests sophisticated honeypot frameworks by combining modern technologies and flexible strategies. These frameworks highlight how important honeypots are for protecting vital infrastructures, improving threat intelligence, and offering a strong, proactive defence against the intricacies of contemporary cyberattacks.*

## 1. Introduction

A honeypot is a network or decoy system intended to attract cybercriminals, imitate legitimate systems, and obtain information about their devices. It serves as a trap, enabling security professionals to research possible dangers without harming vital systems [2]. However, no single technique offers complete security, and there are always several ways to lessen the same threat. By improving the identification and mitigation of sophisticated cybersecurity assaults, adaptive honeypot clouds bridge this gap by changing with threats [5]. Honeypots were first introduced in 1993; they have changed to reflect the new threat [10]. Technologies have changed dramatically, moving from simple traps to complex systems that can imitate whole networks and provide in-depth knowledge on sophisticated cyberthreats [15]. Cybersecurity systems provide honeypot and honeynet configurations to attract intruders [33]. Previous research highlights that the famous Chinese general Sun Tzu, based on Honeypot: "knowing one's enemy." Clifford Stoll first

presented this idea in his book from 1990 book. Honeypots are now a useful tool for forensic study of security incidents and near real-time monitoring. The term "information system resources whose value lies in being attacked and probed" is used to characterize honeypots in the literature [37].

Deploying honeypots is mostly done to track the infection process, find malware, and examine the full picture of botnet activity from the perspective of the infected hosts. To detect and examine malware and other malicious activity, honeypots are essential. IP addresses, commands typed onto the system, and timestamps are among the data gathered by the honeypot [10]. Using deception as a defense mechanism is not an unfamiliar concept; it has historical roots in military strategy, when feints and decoys misled opponents.

Within the field of computing, the idea of computing started to take shape in the 1980s. However, the word "honeypot" itself was not used until much later. Although the primary purpose of these early decoy systems was to divert or slow down invaders, they were basic and set the stage for more advanced strategies [13]. Security has moved from being a secondary factor to becoming a key priority as the Internet's popularity continues to rise daily. Using technologies like client phishing sites and server honeypots, this study presents an integrated architecture for malware collection and analysis [41]. The two types of security breaches are internal (attacks from within the organization) and external (attacks from outside the organization) [50]. The workflow and features of the Honeypot architecture are the main emphasis of this paper's overview of botnet architectures. We conduct a thorough state-of-the-art investigation to provide a more comprehensive picture of how technologies are employed over time [62].

Another significant botnet is HoneypotNet; the first defence technique to employ against backdoor assaults on substitute models is HoneypotNet, which targets attackers who are trying to extract the victim model. This procedure maintains the model's initial performance while altering the output to be malevolent. Results from experiments on four well-known benchmark datasets demonstrate that HoneypotNet successfully inserts backdoors into replacement models. These backdoors serve as a powerful deterrent against model extraction assaults by enabling ownership verification and interfering with the operation of replacement models [56].

Distributed Denial of Service (DDoS) is categorized into two techniques. A harmful technique known as "traffic flooding" overloads network traffic with large amounts of data, preventing network traffic from legitimate clients from entering the network system. Another attack approach is called "Request Flooding," which involves flooding a network service offered by a host or server with numerous requests, making it impossible for the service to handle the requests from a genuine client. The final attack method involves blocking a valid client's communication with a host or server through a variety of means, such as changing system configuration data or even physically damaging the component and server [24]. As the fundamental technology utilized to battle this attack type, DoS/DDoS detection and mitigation techniques have garnered a lot of scholarly attention. These mechanisms include threshold mechanisms, statistical methods, machine learning mechanisms, and multi-method combination mechanisms [2].

## 2. Literature Review

Every year, more and more cyber threats are aimed at vital resources in governmental, commercial, and private networks. Additionally, these dangers reach various versions that are more complex and challenging to identify due to their enhanced harmful properties. For instance, the existence of the Mirai botnet was initially identified in 2016. Mirai, one of the most significant botnets, was initially discovered in August 2016 [3]**.** The adversary can use Mirai, an army of bots, to cause a Distributed Denial of Service (DDoS) attack against the devices in an Internet of Things (IoT) network.DoS/DDoS attacks, also known as East-West bound attacks, are malicious attacks that target the link that connects these controllers [2]. It involves attackers flooding IoT ecosystems or smart city network infrastructure with excessive traffic, overloading it, and making services unavailable to users [29]. A distributed honeypot system provides important information on how effective and resilient it is against various cyberthreats [57].

### 2.1 Functionality of Honey Pots

Honeypots employ clever tricks to avoid attackers. They are intended to draw in attackers by providing

false but alluring services and information. Attackers believe that honeypots contain genuine and useful resources. Nevertheless, these are fake, and the primary objective is to observe and comprehend the attacker's actions. The three primary purposes of honeypots are to detect threats, prevent them, and gather information about attacks.

**2.1.1 Detection:** Honeypots have a low false detection rate since they are excellent. Since legitimate users do not interact with honeypots, erroneous detections are not dangerous. Traditional technologies (such as firewalls and intrusion detection systems) are not very efficient in detecting zero-day attacks, which are detected via honeypots.

**2.1.2 Avoidance (Preventing Attacks)**
Honeypots use three main strategies to stop attacks:
1. Delays attackers so that legitimate systems have more time to protect themselves.
2. Making attackers feel threatened even in the absence of real security measures.
3. Wasting the time, energy, and other resources of the attackers. Figure 1 shows honeypot strategies.


**Figure 1: Honeypot Strategies**

**2.1.3 Study (Comprehending Attacker Conduct)**
Honeypots collect comprehensive information about the actions and responses of attackers. This data is used by researchers to examine the trends of attacker behavior. These patterns enhance overall defence systems by assisting in the identification of adversary tactics and strategies.

Six essential functioning criteria to help administrators and developers choose honeypots. ImCo (implementation cost), DeCo (design complexity), CoRi (compromising risk), CoDa (collected data), and DePo (deception power).Following Figure 2 is shows the functioning criteria of Honeypot,

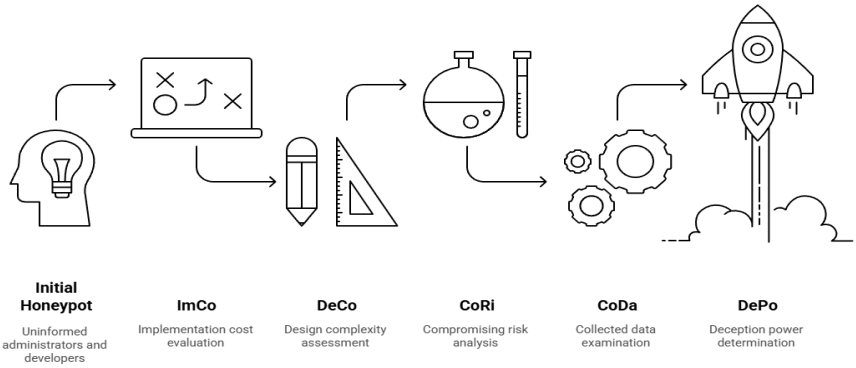



**Figure 2: Six Functioning Criteria of Honeypot**

### 3. Honeypot Techniques

This redirection method is a sophisticated application of security based on honeypots. The attacker is taken to a phony server and cut off from the actual one. The attack is analyzed in real time without jeopardizing any of the original systems. This is an extremely effective tool for attack forensics and cybersecurity defence.

**Table 1: The following are the Honeypot techniques**

| Technique Name | Category |
|---|---|
| Deceptive technique | Deception |
| Advanced Mimicking Technique | Deception |
| False cooperation strategy | Deception/Strategic |
| Subtle disruptions | Disruption |
| The Honeytoken bait method | Deception/Honeypot |
| Technique of traffic redirection | Redirection |
| Mathematical simulation and a model based on social networks | Analytical /Simulation-based |
| A sophisticated method | General /Advanced |
| Methods for VM cloning | Cloning |
| Cloud –based technique | Deception |
| Honey-X techniques | Cloud Computing |
| HoneyCloud | Cloud-Based |
| Honeyweb | We-Based |
| Honey-X-based | Honeypot |
| Honeyfarm is a technique | Honeypot/Farm-based |
| Honeytokens | Token-Based |
| Shuffling-based MTD techniques | Moving Target Defence |
| Diversity-based MTD techniques | Moving Target Defence |
| Redundancy-based MTD techniques | Moving Target Defence |
| CYDEC3 techniques | Hybrid Defence |
| Deception techniques is redirection | Deception |
| Moving target defense is a proactive defense mechanism | Moving Target Defence |
| Hybrid defense strategies using honey-X | Scheduling/Trust-Based |
| Based, including GT-based, Machine Learning, and MTD approaches | Defence |
| OSSSA is a full and trustworthy scheduling mechanism | Game Theory, Machine Learning |

### 4.Types of Honeypot

There are several varieties of honeypots, each with special qualities and uses. Selecting the appropriate kind of honeypot is an important choice that necessitates careful consideration of several important aspects. The network's current condition is the main factor to be taken into account. Since some honeypot types are more appropriate for particular network configurations, it is crucial to comprehend the network's size, structure, and vital resources.

#### 4.1 High Interaction Honeypot

Honeypots are a useful tool for figure out strategies, methods, and procedures (TTPs) of adversaries that attack infrastructure. Important details about attackers, such as their IP address, assault timing, compromise techniques, and instructions they employ to stay persistent, can be found in a comprehensive high-interaction honeypot. For example, when systems are emulated or attackers engage in in-depth contact, maintenance and resource consumption increases significantly [6]. During deployment, forensic data can be retrieved from both high and low

interaction honeypots. The development of low-interaction honeypots came about as the demand for more thorough analysis increased [15]. Every high-interaction honeypot has its own special characteristics. EMPHAsis, for instance, is appropriate for a variety of circumstances due to its adaptable and extendable design. Some high-interaction honeypots incorporate real vulnerabilities to give them a more realistic appearance [28]. For example, an LDAP honeypot was updated with a Log4j vulnerability, which enables hackers to use LDAP to attack directory services. By using this technique, information about how hackers take advantage of this vulnerability was gathered, demonstrating that creating honeypots with actual vulnerabilities can be an effective technique [26]. High-interaction honeypots acquire detailed information about attackers' approaches, tactics, and procedures, making them invaluable for in-depth research [15].

Docker depends on three fundamental technologies: the file system, cgroups, and namespaces. The suggested high-interaction honeypot approach makes use of Docker containers, which have features that make using them in a honeypot easier. Combining Docker with other open-source technologies makes it an effective tool for both preventing honeypot detection and tracking the activities of attackers at the host and network levels [44].

High-interaction honeypots offer more insight into the behavior of the attackers than low-interaction honeypots, but they also carry greater hazards, such as the possibility of attackers compromising the honeypot system itself [36]. The Honeynet is a well-known high-interaction honeypot that enables researchers to observe intricate attacks in a safe setting [13]. Extended attacker involvement through high-interaction honeypots allows for in-depth examination of their tactics. Nevertheless, existing honeypot-based techniques frequently depend on static deployments or are unable to dynamically adjust to emerging threats. Furthermore, complete knowledge of attacker methods is often assumed by game-theoretic models used for attack detection, which is an unrealistic assumption in practice scenarios [53].Deploying a high-interaction honeypot in an Industrial Control System (ICS) context requires using an actual Programmable Logic Controller (PLC) or other ICS device. These devices are costly, and an ICS deployment is not accurately represented by a single device. Therefore, it is necessary to install several devices in order to transport data between them to achieve complete high-interaction [52].Following Figure 3 shows the Honeypot Interaction Level.

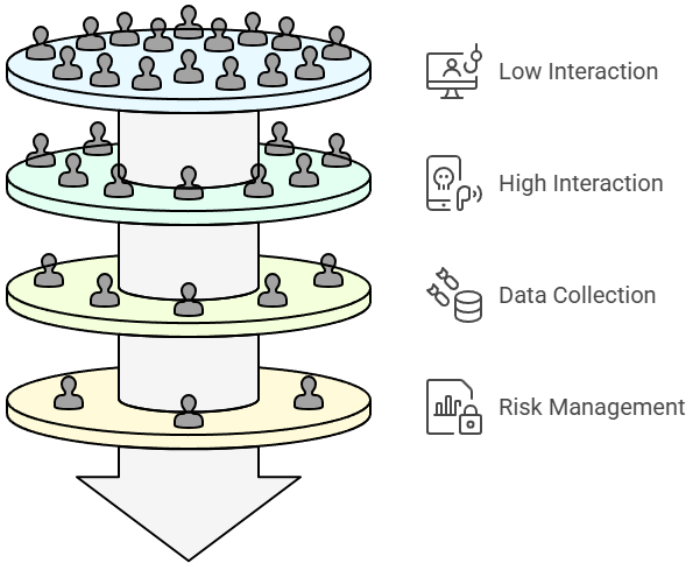



**Figure 3: Honeypot Interaction Level**

### 4.2 Low Interaction Honeypot

Low-Interaction Honeypots: These are simple and offer little utility. They need minimal upkeep and are lightweight. Examples are DTK and Honeyd, a low-interaction honeypot created in the early 2000s that allowed users to construct network services using open ports and fake IP addresses [13]. Low-interaction honeypots only simulate a limited number of services, such as SSH or FTP, and they deny the attacker any access to the operating system. One important low-interaction client honeypot that is skilled at identifying server-based assaults with the least amount of sophistication is HoneyC [26]. This only mimics simple protocols, such as FTP and SSH. Their answers are usually confined to handshake contacts, and they limit access to the underlying operating system [55]. Dionaea and Honeyd are examples of low-interaction honeypots that mimic a variety of services to draw in and examine malware; Dionaea is made to capture exploits that target services like SMB, HTTP, and FTP, while Honeyd can create virtual hosts on a network that mimic different operating systems and network configurations.

### 4.3 Distributed Honeypot

As cloud-based and distributed architectures have grown in popularity, honeypot designs have changed to accommodate new settings. With the use of distributed honeypots, businesses can set up honeypots on several nodes, frequently dispersed throughout several regions [13]. Distributed honeypots are set up in many locations, expanding the attack surface and improving the likelihood of identifying a variety of threats. This makes them useful for monitoring and analyzing widespread attack patterns by major organizations and internet service providers (ISPs) [15].

Following Figure 4 shows the Distributed Honeypot Architecture.

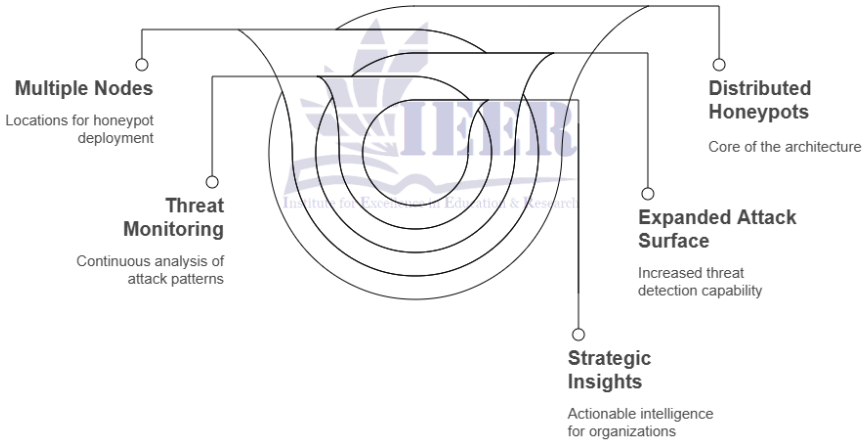



**Figure 4: Distributed Honeypot Architecture**

### 4.4 Hybrid Honeypot

Low-interaction and high-interaction system components are combined in hybrid honeypots. These honeypots provide a balance between analyzing certain threats in-depth and recording large amounts of attack traffic [36]. As businesses look to implement effective yet efficient security measures without incurring the resource overhead of monitoring numerous systems, hybrid honeypots have grown in popularity [13].

### 4.5 Network Honeypot

Network honeypots support other security technologies to identify and assess network risks. They are appropriate for organizations that want to strengthen current security protocols and offer early warning of possible attacks [15]. Honeypots provide a variety of cybersecurity functions, including machine learning-based SSH brute-force attack classification [10]. Furthermore, recent research has shown how honeypots can be integrated with other security systems. Developing complex systems capabilities has been linked to integrating honeypots with blockchain

frameworks and machine learning techniques, enabling them to anticipate and mitigate cyberthreats [12] more effectively. Intrusion Detection Systems (IDS) for Linux and Windows were implemented using two different kinds of honeypots, Kfsensor and Honeyd, respectively. Initially, traffic can go via a Honeywell, which can be connected to an Intrusion Detection System (IDS). IPTables can also restrict connectivity between the actual systems and the honeypots [3]. Honeypot integration with intrusion detection systems produces positive outcomes [41]. Following Figure 5 shows the Distributed Honeypot Architecture.

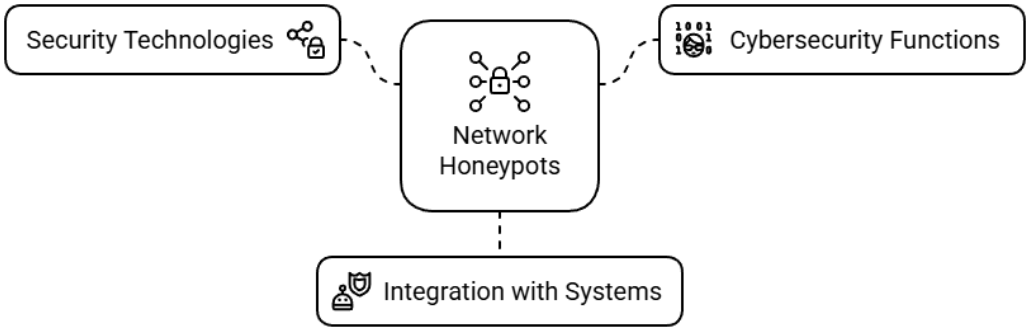



**Figure 5: Distributed Honeypot Architecture**

### 4.6 Virtual Honeypot

Virtual honeypots can be configured to operate on any platform, including cloud servers and personal computers, because they are not restricted by specific physical hardware [18]. Because virtualization technology is so sophisticated and there are so many software packages available, virtual honeypots are a typical occurrence in traditional computer security. There are both real and virtual honeypots on the LAN. A physical honeypot is used to simulate a vulnerable host in order to draw hackers, and a virtual honeypot is used to hide the Demilitarized Zone (DMZ). In addition to reducing the danger of server attacks, the virtual honeypot can save money.

### 4.7 Medium Interaction Honeypot

Between a low-interaction and high-interaction honeypot, a medium-interaction honeypot compromises some operating system validity to facilitate data analysis [28]. Organizations commonly utilize medium-interaction honeypots as a compromise since low-interaction honeypots have poor data quality and are unable to swiftly assess the wealth of information offered by high-interaction honeypots [52]. However, these honeypots are easy for adversaries to find, which reduces the quality of the collected data. Understanding malicious behavior after acquiring access is the main goal of medium-interaction honeypot analysis. As a result, while a broad summary is still provided, temporal analysis is given less weight than in the prior subsection [55].

### 4.8 Cloud Honeypot

A cloud security tool called a cloud honeypot is placed in the cloud to attract and record attacks directed at cloud systems. It appears to be a cloud service that is vulnerable, like storage or a virtual machine, to fool attackers. To investigate how attackers target cloud systems, researchers employ cloud honeypots. Data breaches, illegal access, and security flaws are all detected with their help. Through the analysis of these attacks, security teams may enhance cloud security. Cloud service providers can improve system security by using cloud honeypots. They provide important details to strengthen cloud security.

In addition to categories based on interactions, honeypots can be grouped according to their function within a security ecosystem:

1. **Production Honeypots [ReH]**

Positioned thoughtfully throughout a company's network to strengthen defenses. These honeypots, which are often low-interaction systems designed to catch less complex attacks, are centred on early detection.

2. **Research Honeypots [PrH]**

These are set up by security researchers to gather information on widespread or unusual attack patterns. For the advantage of the security community, these systems usually permit high interaction to collect comprehensive hostile TTPs [31]. Following Figure 6 shows the Cloud Honeypot Security Ecosystem.

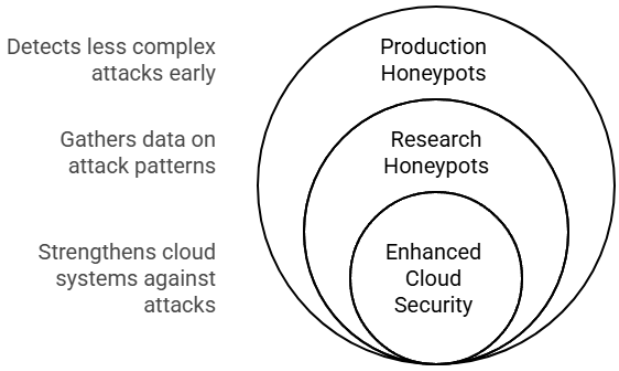



**Figure 6: Cloud Honeypot Security Ecosystem**

The suggested solution was determined to be appropriate for network security after the examination of both honeypots, serving as a further line of defence to identify malicious activity anytime it occurs on the network [29]. Security experts claim that honeypots fill in a number of the holes left by conventional intrusion detection systems (IDS) [38]. Many researchers use intrusion detection systems and honeypot technologies to come up with answers and propose ways to deal with the many problems that intrusion detection systems encounter. Similarly, a signature generator has been suggested as a way to improve digital network security through the use of honeypot technology [41]. Furthermore, cybersecurity systems and computer (host) security systems make up cybersecurity systems. At the very least, each of these has an intrusion detection system (IDS), a firewall, and an antivirus program. Unauthorized usage, duplication, change, and destruction of information systems are detected, identified, and determined with the aid of IDSs [1]. In order to speed up the scan, Mirai bots send TCP SYN packets without finishing the three-way handshake to identify additional vulnerable targets. Furthermore, the Mirai source code shows that hackers use a signature to carry out their actions. During Mirai botnet searches, the TCP sequence number is set to the IP address of the corresponding author. Destination (TCP. seq == IP.dst). Our starting point is this signature, which we refer to as the Mirai signature.

Additionally, by examining the TCP SYN packets that confirm the Mirai signature, we look at how Mirai botnet scans have changed over six years, from 2016 to 2024[62]. Following Figure 9 shows the TCP SYN Packet Trends Across Ports.

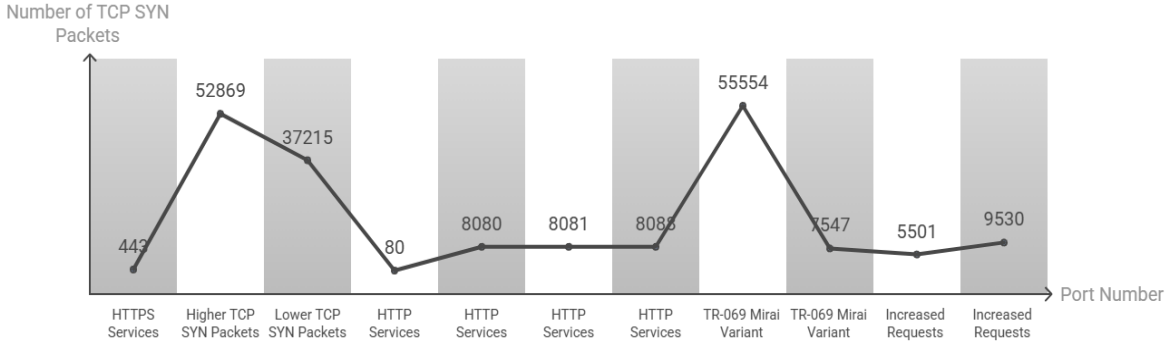



Figure 9: TCP SYN Packet Trends Across Ports

There has been a current initiative in both academia and industry to investigate the application of deception techniques to enhance proactive attack detection and defense. The intentional activities done to deceive hackers and make them take (or not) particular actions supporting computer security defenses are known as computer security deception operations. Researchers studying computer security have looked into how hackers use deceit to attack networks and how misleading honeypot devices are employed to protect them. On the other hand, not much has been done to model and analyze computer security deception operations systematically [64]. The fundamental idea of the DCD concept is to use deception resources like Honeypots[65][8][14], Honeytoken[60],Honeyfarm[15][30]suspicious internet traffic to a specialized gateway, which examines it before forwarding it to virtual machines for additional analysis Honeycomb, Honeybrid[30] and Honeynet[62][21] to entice and trap attackers by providing realistic but misleading information. Although honeypots and honeynets were not designed with the Internet of Things in mind, they are being used in research for these devices[21].

A popular open-source framework called Honeyd is used to build scalable, low-interaction honeypots that mimic popular network protocols and services. A honeypot system with low-to-medium interaction is called Honeyd [65]. When installed on a UNIX system, it watches for incoming ARP queries on the network interface card (NIC). Honeyd starts an ARP request on its own if one is identified. Another lightweight and scalable honeynet framework designed for IoT settings is called TrapNet. Using micro services for efficiency, it combines local, little honeypots with more expansive, adaptable honeynets housed on cloud and fog platforms [54].

**Table 2: Honeyd fields test Results**

| Events | No of Occurrence |
|---|---|
| Nimda | 8871 |
| CodeRed | 2155 |
| CodeRed II (3 versions) | 2629 |
| MyDoom | 1369 |
| W32/Welchia.D | 1674 |
| Attempts to access the IIS-Samples | 645 |
| Attempts to get '/ect/passwd' | 168 |
| Attempts to execute cmd.exe | 12345 |

Because of their limited interaction capability and ability to mimic several decoy systems, they are frequently employed to identify the existence and volume of attacks [4].The idea of deception was initially described as a user account filled with many fabricated files to entice and delay an attacker while being monitored. Eventually, this made-up setting intended to record attacker contact was dubbed a honeypot [3]. Deception systems act as both attack targets and information gathering tools. One of the features of a deception system is the ability to monitor an attacker's behavior by allowing them access. This crucial feature of the deception system also determined their course of action.

**5. Adaptive honeypot technologies and hybrid cyber defense strategies in modern cybersecurity**

Researchers and industries are using honey X-based deception techniques such as honeypots, honeynets, honeytokens, and honeywebs. To increase the efficiency of cyber defense, several academics have put together several hybrid defense strategies that combine cutting-edge DCD Deceptive Cyber Defense) and Moving Target Defense (MTD) techniques with new developments in machine learning (ML) and game theory (GT) [4]. Cryptography, firewalls, and antivirus software are common security methods that are used, with some modification, to safeguard services [9]. Mirai establishes a network of malware-compromised devices, including routers and security cameras. A botmaster can remotely manipulate these

compromised devices, often known as "bots," to do harmful tasks [62]. One of the fascinating topics in cybersecurity is the use of an adaptive honeypot as a cyber defence weapon. The machine learning algorithm and honeypot combination are chosen to accomplish two distinct objectives: to interact with the attacker and to prevent the possibility of being compromised. Using an adaptable honeypot also has the advantage of being able to evade detection by popular survey and honeypot detection technologies [10]. Adaptive honeypots, which are made to change their behavior in response to the activities of attackers, have proven to be very successful. By using reinforcement learning to dynamically modify answers, these systems enable more efficient attacker interaction while lowering the chance of detection [12].Different botnet architectures allow botmasters to interact with their bots in different ways. Although each structure has its pros and cons of its own, the major question is still whether the botmaster can interact with the botnet covertly. There are numerous methods for identifying botnets. The security community finds great value in honeypots among these. Honeypots help defenders detect and prevent possible botnet attacks when used in combination with other security solutions [47]. Following Figure 10 shows the Cyber Defense Strategies and Technologies.

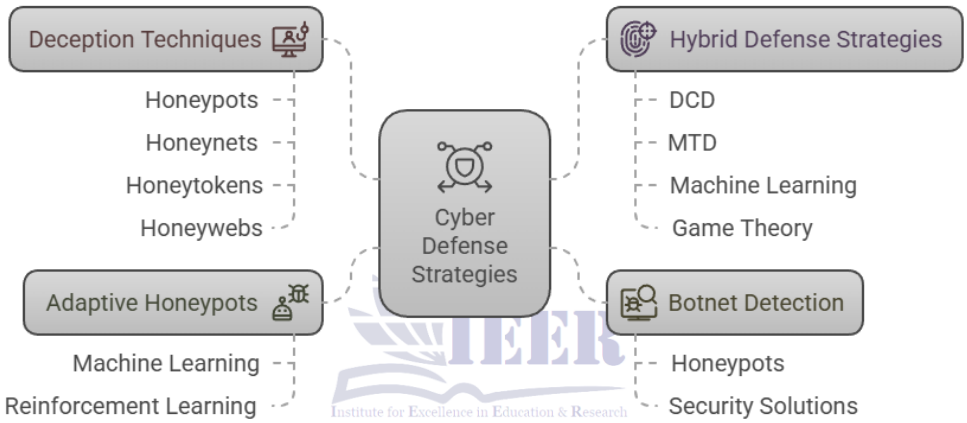



**Figure 10: Cyber Defense Strategies and Technologies**

Adaptive systems like ASGuard, which engage attackers through reinforcement learning while being shielded against severe compromises, serve as more examples of the idea. ASGuard demonstrates the efficacy of reinforcement learning in reducing cyber risks by optimizing honeypot functionality by developing reward functions that strike a balance between attack data collecting and system safety [12]. SSH, Honeypot can be merged with the DDQN algorithm. Each action is more optimally determined using the DDQN algorithm because it calculates two Q-Values, as seen by how it reacts to particular requests. In addition to using the memory server less frequently than the DQN honeypot, DDQN can assist the honeypot in learning more quickly than DQN. Another technique that can be combined with the honeypot is the DDQN algorithm because it uses less memory than the DQN algorithm [10]. Behavioral analytics, Endpoint Detection and Response (EDR) tools, and Security Information and Event Management (SIEM) systems are revolutionizing how businesses identify and address cyber threats [1]. The standard evaluation model is used when evaluating intrusion detection systems. Following the deployment of every candidate IDS, each IDS examines a benchmark dataset, and a selection of metrics is used to grade the accuracy of its classifications [5]. Numerous types of honeypot software have been created, and they can be categorized as low, moderate, or high interaction levels [4]. A subset of the AAA infrastructure called Multi-Factor Authentication (MFA) is being utilized more and more for safe user authentication and authorization of crucial network infrastructure devices. The organization's requirement for a consistently operational connection between its

infrastructure and the AAA/MFA service provider is increased by the centralization of the AAA and MFA, particularly when employing verification techniques that require Internet/network access. Some organizations do not tolerate this kind of dependence. Using third-party networks and infrastructure to conduct AAA/MFA communication is also not recommended [6].Numerous combinations in several recent studies that adopted hybrid defence strategies. Clark et al developed a decoy-based deception (belonging to DCD) and IP randomization defense mechanism (belonging to MTD) against scanning attacks [9]. Later, Yuyang et al. used MTD and cyber deception techniques to create a hybrid proactive defence system against Distributed Denial of Service assaults in the Internet of Things area. Then, using their own open-source Software-defined network (SDN) platform, Mengmeng et al. tested their defence system, which combined deception and MTD tactics in a smart hospital scenario [4].

**Table 3: Deception Techniques and their evaluation Metrix**

| Technique Name | Evaluation Metric | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Deception Discrepancy | Launched Attacks | Returned Adversaries | Second Session | Wasted Time | Using Ration | Traffic Volume | Confusion Matrix |
| Advanced mimicking | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ |
| Fake Cooperation | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ | ✗ | ✓ |
| Deceptive Database | ✓ | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ |
| Subtle Interruptions | ✗ | ✗ | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ |
| Honeytoken Bait | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ |
| Traffic Redirection | ✗ | ✗ | ✓ | ✓ | ✓ | ✗ | ✓ | ✗ |

One of the first steps in preventing cybercrimes is anticipating cyber threats, which lays the groundwork for other components (Nicholls et al., 2021). A well-known real-world example to support the aforementioned is the 2020 SolarWinds Supply Chain Attack, which had an impact on numerous organizations worldwide. Analytical prediction might have prevented the assault by detecting weaknesses before their exploitation (HIMSS, 2021) [9].

There have been numerous studies on adaptive honeypots, some of which are listed here. Using the Pybrain Library, Pauna and Bica combined the SSH DDOS Honeypot Kippo and RL algorithms to produce RASSH, which combined the SARSA algorithm with the Markov decision process [10].

When Pauna et al.combined the Cowrie Honeypot with the DQN Algorithm to produce QRASSH, they carried on the research. Suratkar et al connected the Cowrie honeypot with the DQN Algorithm to deceive the attackers' honeypot detection tools and stop them from utilizing them [10]. The idea is to combine honeypots with reinforcement learning algorithms to create adaptive honeypots. Researchers have presented the idea of an adaptive honeypot, which can learn from the actions of attackers. Because it allows for longer interactions between honeypots and attackers, the adaptive honeypot is one of the most fascinating ideas in cybersecurity [31].The use of an adaptable honeypot as a cyber-defensive weapon is one of the most interesting subjects in cybersecurity. Combining a machine-learning algorithm with a honeypot is used to achieve two different goals: communicating with the attacker and avoiding potential compromise. The ability to avoid detection by widely used survey and honeypot detection technology is another benefit of using an adjustable honeypot [10].

Researchers can better understand and counter new risks, such as sophisticated cyber threats and zero-day

vulnerabilities, by capturing and examining malicious activity [12].

Large language models (LLMs) have been used to create more dynamic and accurate honeypots. By offering thorough insights into human adversaries' strategies and tactics, these technologies can engage them more successfully. Additionally, the implementation of Investigations on honeypots in wireless networks has yielded significant advantages. Wi-Fi honeypots improve security for residential and business networks by quickly identifying and reacting to unauthorized access [12]. Attack detection systems have been designed and simulated using virtual honeypots. These systems have established themselves as vital resources for network security by analyzing large datasets and identifying cybersecurity flaws [12].The Indicator of Attack (IoA) approach is used to present cybersecurity threat intelligence rather than the widely utilized Indicator of Compromise (IOC) methodology. IOC uses forensic data gathered following a cyberattack to determine the attackers and comprehend how the assault occurred. Since our study's main goal is to identify threats and notify administrators of any efforts to penetrate SME networks [14].

Similarities to spam honeypots are seen, although malware honeypots are primarily concerned with examining dangerous software that targets organizational systems. Honeypots and other settings are useful tools for studying the behavior of bots and other automated threats specifically made to draw in these kinds of entities. These technologies play a major role in identifying and stopping bot-based invasions and undesired automated traffic [12]. Although the high degree of freedom in the system makes the power grid more convenient, Industrial control systems (ICSs) are now a crucial component of the nation's vital infrastructure, including gas pipelines, power grids, and even aircraft. With Industrial Control System (ICS) equipment like as Intelligent Electronic equipment (IEDs), Programmable Logic Controllers (PLCs), and Remote Terminal Units (RTUs), computer networks are modernizing traditional industry.

ICS devices to transmit data, although this convergence may give birth to additional security issues [18], use the Transmission Control Protocol and Internet Protocol (TCP/IP) stack**.** However, in general, ICS security experts do not frequently consider honeypots. Standards, rules, and regulations are some of the primary forces behind the security of ICSs and critical infrastructure. In an ICS environment, standards such as ISO 27019, IEC/ISA 62,443, and NIST SP 800-82 are typically utilized. As a result, industry uses these documents for ICS infrastructure deployment and security [52]. Many ICS honeypots can be easily recognized with minimal interaction using basic networking tools. A study of over 8,000 devices claiming to be ICS systems highlights this issue [58]. It also exposes the grid to several cybersecurity risks, including Rather from being taken over by the system right away. APTs use a variety of techniques to continuously and secretly compromise the target system to examine the long-term relationship between the cyber and physical layers [17].APT can be divided into two stages: intrusive and disruptive, when compared to other threats. The attacker initially breaches the system to determine security during the harmful stage, after which it keeps gathering data and sends it to the attacker [4]. When the APT attacker has identified the security tools, the physical equipment that they are protecting is vulnerable to damage during the disruptive stage [17]. Cyberattacks on ICS devices have been growing more frequent and damaging in recent years [18]. Under N-day APT, we examine the effect of subjective parameters on attackers' payout. Attacks by calculating UW1's partial derivative of $Uw1(v1)to\ \lambda\ \alpha$ and $\beta$ respectively.

$\partial Uw_1(v_1)/\partial\lambda = -m/(m+n)(1-w^A(p_2)) = -n/(m+n)(1-w^A(p_1)).\gamma_1{}^{\beta}$

$(\partial Uw_1(v_1)/\partial\alpha) = \Sigma\ (p = 1\ to\ m)\ (1/(m+n))\ w^A(p_2)\ (\varepsilon_2,p-\gamma_1)^a\ ln(\varepsilon_2,p-\gamma_1)$

$+\ \Sigma(k = 1\ to\ n)(1/(m+n))\ w^A(p_1)(\varepsilon_1,k-\gamma_1)^a\ ln(\varepsilon_1,k-\gamma_1)$

$\partial Uw_1(v_1)/\partial\beta = -m\ (1-w^A(p_2))\ \lambda\ \gamma_1{}^{\beta}\ ln(\gamma_1)$

The following conclusions could be drawn from an analysis of the functions that have been mentioned.First, since $1-w^A(p_i) > 0$ is valid, it is evident that $\partial Uw_1(v_1)/\partial\lambda < 0$ always holds true. This implies that the attacker will progressively underestimate his or her payment as $\lambda$ rises.

Second, the difference between $\varepsilon_1$, k, $\varepsilon_2$, p, and $\gamma_1$ determines the monotonicity of $Uw_1(v_1)$ to $\alpha$. $\partial Uw_1(v_1)/\partial\alpha < 0$ is the result of $0 < \varepsilon_1,k-\gamma_1 < 1$ and $0 < \varepsilon_2,p-\gamma_1 < 1$. Consequently, there are no deterministic consequences and $\partial Uw_1(v_1)/\partial\alpha > 0$ is

the result of $\varepsilon_1$,k, $\varepsilon_2$,p - $\gamma_1$ > 1 and $\varepsilon_1$,k - $\gamma_1$ > 1 and $\varepsilon_2$,p - $\gamma_1$ > 1.Finally, the difference of $\gamma_1$ determines the monotonicity of $Uw_1(v_1)$ to $\beta$. There are no deterministic outcomes for other conditions. Since 0 < 1 - $w^A(p_i)$ < 1 always holds, 0 < $\gamma_1$ < 1 results in $\partial Uw_1(v_1)/\partial\beta$ > 0 therefore, $\gamma_1$ > 1 results in $\partial Uw_1(v_1)/\partial\beta$ < 0 without a doubt [17].

Two main categories of methods are used to predict an assault phenomenon: quantitative and qualitative. Professionals typically develop their ability to address cyberthreats by simulating real-world difficulties through intense, practical training sessions known as cyber defence exercises (CDX) [19]. Cyber threat anticipation tools' predictions and trend analysis can be strengthened by integrating threat intelligence feeds. When combined and connected with current cyber threat data, real-time patterns of interest can be found through continuous monitoring and data analysis, which could help businesses forecast their activities [9].Following Figure 11 shows the Predicting Cyber Assaults using quantitative and qualitative methods.


**Figure 11: Predicting Cyber Assaults using quantitative and qualitative methods**

It has long been a challenge to predict the features of a cyberattack, and the various approaches that have been tried produce varying outcomes based on the kind of attack [9]. You may update your knowledge base on firewalls and intrusion detection systems, discover the newest attack techniques and tools, and discourage the intruder to some extent [16]. Honeypots can be combined with firewalls and intrusion detection systems to create an intrusion prevention system (IPS), which can gather all the data about attackers, analyze all of their activity, and create strategies to strengthen system security and stop future attacks [21].

Intrusion detection and firewalls have been applied extensively and are crucial components of LAN security systems [16].The following are the functions of a honeypot, which is intended for active defense: It may attract hackers to target it and safeguard the actual objective; It can capture and securely preserve the evidence; it can capture and pinpoint the goal of hacker attacks as well as their tactics and equipment [16]. A DMZ is a logical or physical subnetwork that divides an insecure external network, usually the Internet, from an internal local area network (LAN). By separating publicly available resources, like web servers, email servers, and DNS servers, from the internal network, a DMZ serves to secure an organization's network further.This configuration guarantees that the internal network is safe even if the DMZ resources are hacked. The attackers will quit the server attack and leave once they realize they are targeting a honeypot and the network has already set the trap. Since many skilled hackers will not easily quit the attack, physical honeypot deployment is done in this system to achieve improved cover-up to gather information about different attacks. Numerous

physical computers will be set up as standard personal computers, and the operating system of the trap host will have numerous security flaws purposefully placed in place before being closely watched.

## 6. General Applications

Numerous honeypot technologies with low to medium engagement levels provide distinct features suited to various security requirements. Key open-source honeypot apps and their features are examined below:

### 6.1 Honeyd

Honeyd is an open-source software for the creation of low-interaction honeypots. In addition to creating virtual honeypots, Honeyd enables the integration of physical machines [26]. It is capable of simulating Telnet, IIS, FTP, SMTP, TCP, UDP, and POP services [21].To ensure that the simulated systems react to the three main IP protocols—Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and Internet Control Message Protocol (ICMP)—it replicates the network stack for the computers[23]. As a flexible deception and cybersecurity research tool, Honeyd can simulate a wide range of operating systems, services, and network settings. Scaling for a wide variety of IP addresses is possible with Honeyd [30].

### 6.2 Dionaea

An open-source program called Dionaea [26] can be used to create medium interaction honeypots that can mimic a number of services, including FTP, HTTP, MongoDB, MQTT, MySQL, SIP, SMB, TFTP, UPnP, and others [59]. It attacks advertisements that target hosts with weak services on the Internet. Dionaea seeks to acquire a copy of malware and assist researchers in analyzing it, as adversaries attempt to install malware on the compromised machine [36].

### 6.3 Kippo

Kippo is a medium-interaction, scalable, open-source honeypot that mimics a Secure Shell ( SSH) server (n network protocol). Additionally, because it has fewer features, there is less chance that the system itself may have weaknesses, which further improves Kippo's security [26]. Its main objective is to track and record brute force attacks and interactions that are either started by human attackers or automated programs. Because SSH services are the target of so many attacks, Kippo has become more and more popular [28]. Kippo has demonstrated its versatility for a variety of contexts by being updated by researchers like Dowling et al. to execute particular use cases, such as building a ZigBee IoT honeypot. To further illustrate Kippo's value in bolstering cybersecurity defenses, Pauna used it to create the Reinforced Adaptive SSH (RASSH) honeypot.

### 6.4 Adaptive Honeypot Alternative (AHA)

Wagener created the self-adaptive SSH honeypot Adaptive Honeypot Alternative (AHA) by applying machine learning (ML) and game theory. This method involved gathering information from attackers using both low- and high-interaction honeypots, which was then used to construct an adaptable system. Despite not being applied in an Internet of Things setting, Wagener's work served as a basis for Pauna's later research endeavors [28]. According to Wagener's research, attackers used the adaptive honeypot's customized tools three times more frequently, underscoring the significance and potential use of adaptive honeypots in honeypot studies.

### 6.5 Rootkit

Pauna enhanced Wagener's adaptive honeypot in 2012 with AHA with Rootkit Detection. Pauna created a scalable, medium-interaction virtual honeypot that can identify rootkit software that attackers have implanted. Pauna's honeypot uses Argos to identify rootkit malware and runs as a guest operating system (OS) on the Argos emulator. This emulator detects the malware. This enhancement improves the adaptive honeypot's capabilities, enabling it to detect and react to malicious software in addition to gathering information on attacker activity.

### 6.6 Reinforced Adaptive SSH (RASSH)

In 2014, Pauna and Bica developed the adaptable honeypot known as Reinforced Adaptive SSH (RASSH). It has two modules: the Actions module and the Reinforcement Learning module, and it makes use of a Kippo honeypot. Based on the Reinforcement Learning module, RASSH communicates with attackers and performs dynamic

actions like permitting, blocking, or postponing. A self-adaptive IoT honeypot called Intelligent Reinforced Adaptive SSH (IRASSH-T) was created because of this research.

**6.7 Cowrie**

A program called Cowrie is used to build scalable virtual honeypots with medium to high interaction levels. It acts as a medium for interaction honeypot, simulating commands and recording an attacker's actions on a simulated UNIX system. Cowrie's play log is a screen capture in UML file format. Even while the play log can be accessed instance-by-instance, it cannot be used for command extraction or query search. As a result, the writers used a creative method to work with Cowrie's play log [33]. It serves as a proxy for SSH and Telnet to track the attacker's activities on a different system, making it a high interaction honeypot. Additionally, Cowrie can provide flexibility by serving as a conduit between an attacker and a collection of virtual computers. It can mimic services including SSH, Telnet, SFTP, SCP, and TCP/IP and was created based on the Kippo honeypot. Additionally, Cowrie interacts with logging, storage, and visualization tools like as ElasticSearch, LogStash, and Kibana.

**6.8 HoneyPy**

Based on the services it replicates, HoneyPy is software for building low-to-medium interaction honeypots. Numerous plugins are included to duplicate services like Domain Name System (DNS), Network Time Protocol (NTP), and others. Additionally, HoneyPy can be set up to operate with particular settings if necessary. It allows for the usage of other services for log analysis and offers many logging alternatives, such as ElasticSearch, Logstash, RabbitMQ, Slack, Splunk, and Twitter. Metognon and Sadre's IoT honeypot study made use of HoneyPy. Researchers Metagnon and Sadre used honeypots to evaluate the security of the Internet of Things (IoT).

**6.9 QRASSH (Q Reinforced Adaptive SSH)**

SSH, Pauna et al. presented honeypot known as QRASSH (Q Reinforced Adaptive SSH) a novel in 2018. It combines Deep Q-learning and Cowrie methods. They discovered, meanwhile, that QRASSH's reward functions were subjective and thus not always the best. As a result, they recommended greater study to create more precise incentive systems for behavior. Subsequent research on this topic resulted in the development of IRASSH-T, an IoT-specific honeypot.

**6.10 OpenVAS**

One free program for scanning and identifying security issues on a computer or network is called OpenVAS [26]. It began as a component of the Nessus project and was later enhanced and developed independently. OpenVAS finds vulnerabilities in systems and apps by doing scans. Its primary component is a server that scans other machines for security flaws using special software. Nessus is an attack scripting language that allows users to execute various network tests. The framework of OpenVAS is straightforward, and adding new features is simple. Additionally, it features a graphical user interface, which makes it easy to use. However, because of its age, the interface may be more difficult to use.

**6.11 Glastopf**

Glastopf is a honeypot that simulates a web application to obtain information from intruders [26]. Common attack targets include web applications, databases, and cross-site scripting flaws, which can be exploited to perform drive-by download attacks, propagate spam emails, destroy websites, and build website bots. Glastopf is a low-interaction honeypot, which means it imitates weak web servers with thousands of flaws in several web pages and applications, drawing in attackers and gathering data about their activities.

**6.12 Methods of detection**

1. Detection Based on Signatures: Compares to known attack signatures.
2. Behavior analysis: Tracks how users and systems respond to circumstances [36].
7. Finding departures from typical behavior is known as anomaly detection.

**7. Deployment of web portal Honeypot**

Due to their limited resources, high startup costs, and fully occupied staff, small and medium-sized enterprises (SMEs) frequently struggle to secure their IT infrastructures, which causes them to place less emphasis on cybersecurity. Existing implementations

are mostly made for large firms with specialized cybersecurity teams who can analyze the data and put countermeasures against new threats, even though cybersecurity threat intelligence from honeypots might help businesses reduce risks [38]. The frequency of cyberattacks on SMEs has significantly increased, especially during the COVID-19 pandemic, and attack methods have become more sophisticated. Ransomware is being used by organized groups of threat actors to deliberately attack the IT infrastructures of SMEs, targeting companies in industries like manufacturing, finance, education, and event organizing.

Setting up a fake web application that looks like a real-world portal but purposefully has flaws to draw in potential attackers is known as web portal honeypot deployment.

An outline of the deployment procedure is provided here:

### 7.1 Build the Web Portal

Create a web application that replicates popular web services that SMEs utilize, such as an admin panel, login page, or online store. The program should contain purposefully unsafe elements, such as out-of-date software, inadequate authentication procedures, and incorrectly configured settings

### 7.2 Simulate Vulnerabilities

Add known vulnerabilities, such as SQL injection points, unpatched security flaws, or cross-site scripting (XSS) problems, to the online portal. Attackers use these weaknesses as bait.

### 7.3 Put the honeypot in a regulated setting

To make sure the honeypot does not affect actual systems, host it in a network that is divided into segments. To avoid cross-contamination, it can be set up in a cloud environment with restricted access or on a virtual computer.

### 7.4 Tools for Monitoring

Use monitoring software to keep tabs on all online portal traffic and interactions.

These tools should record all requests, payloads, attempted exploits, and attack sources.

### 7.5 Threat Intelligence System

Install an analysis system to examine the data that has been collected and find trends in the tools, tactics, and strategies used in attacks. It should be possible for the system to distinguish between harmful and normal user behavior.

### 7.6 Feedback Loop

Provide a way for the organization's security systems to get information from the honeypot. Depending on the risks found, this entails updating vulnerability management procedures, intrusion detection systems, and firewalls.

### 7.7 Continuous Maintenance

To maintain the honeypot appealing to new attackers, update it frequently with fresh attack scenarios and vulnerabilities. It should continue to offer intelligence on current strategies for attack while maintaining isolation from critical infrastructure. Following Figure 12 shows the Deployment of web portal Honeypot**.**

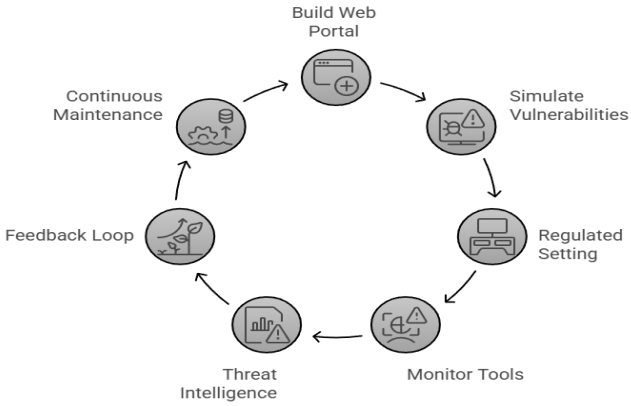



**Figure 12: Deployment of web portal Honeypot**

**8. Deployment of containerized environments using Kubernetes and honeypots**

1. Kubernetes (K8s): A scalable, dependable, and always-available platform for managing container apps.
2. MicroK8s (mK8s): A condensed form of Kubernetes designed for small systems, such as Internet of Things gadgets.
3. K3s: A simplified version of Kubernetes that is easy to use and optimized for low-power settings.
4. Minikube: A tool for studying and testing Kubernetes on your PC.
5. A honeypot system to investigate how attackers target Kubernetes systems is called HoneyKube [39].

**9. Detecting Ransomware and Honeypots' Purpose in Cybersecurity**

Since the late 1980s, cyber-extortion has been practiced. However, in 2005, ransomware became more sophisticated. Ransomware encrypts data and requires payment to unlock it, in contrast to previous attacks that merely destroyed it. This kind of malware is a type of scareware, in which users are coerced into paying out of concern that their data may be permanently lost [35].In recent years, cybersecurity has changed significantly due to more sophisticated threats and attack techniques. APTs and highly skilled ransomware attacks are only two examples of the complex cyberthreats that organizations increasingly face, according to research published in IEEE Security & Privacy [40].

As the ransomware threat changes, new malware variants emerge; some of these are well known, including TeslaCrypt, Locky have just surfaced, along with CryptoLocker, and CryptoWall.It can be difficult to identify malware before it begins encrypting files. To update virus definitions on protected systems, traditional antivirus software must first locate and examine a sample of the infection. Only roughly, half of antivirus applications can protect against a new attack after it has been propagating for a few days. Following Figure 13 shows the Key Milestone of ransomware evolution:

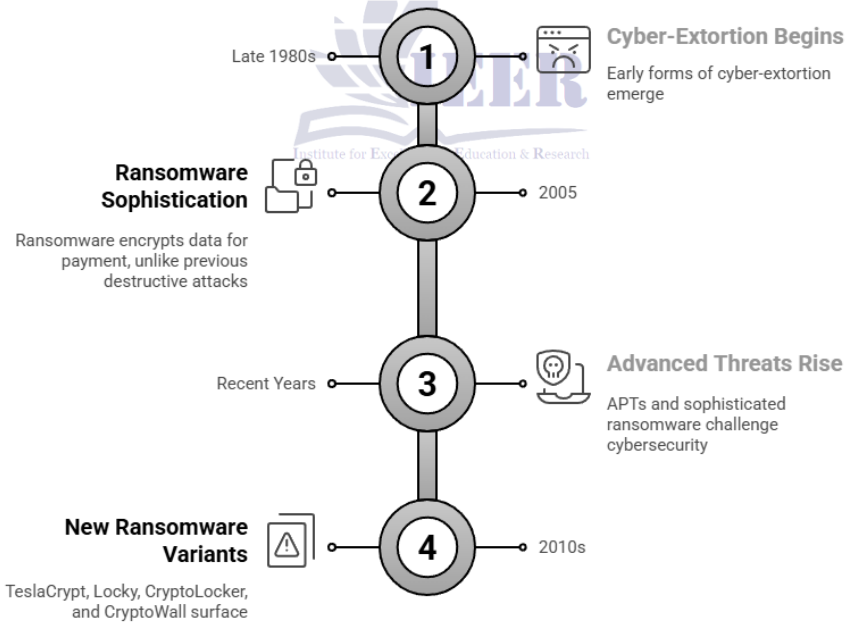



**Figure 13: Key Milestone of ransomware evolution**

As micro services have become more and more widespread, new cyberattacks have been created especially to take advantage of this architecture. Honeypots have shown themselves to be useful instruments for gathering actual attack data and comprehending the strategies employed by attackers. Security professionals can learn more about attack techniques and the kinds of threats that attackers employ by putting up honeypots. However, micro services architectures have not been used well in

traditional honeypot designs, which presents a chance to create honeypots with improved capabilities based on the special characteristics of micro services [39].

It's critical to halt ransomware as soon as it begins in order to stop additional damage because typical antivirus software is slow to identify new viruses. Using a honeypot, a device made to detect unauthorized activity, could be one way to solve the problem. Any engagement with honeypots is viewed as an attack since they do not anticipate actual users. This can assist in warning security professionals about possible dangers.

Examples of security risks to an onboard network include the following [6]

1) Private information could be exposed in the event of an attack on an onshore communication infrastructure. Additionally, the on-shore communication system might be used to attack on-board systems.

2) On-board information systems could be the target of a cyberattack if they are directly connected to onshore systems through a network.

3) Malware sent to emails has the potential to develop a backdoor for external assaults or directly infect on-board information systems.

4) On-board information systems may not function normally if they are subjected to a DoS assault or wireless communication jamming.

### 10. Architecture of Honeypot and its Installation

A centralized Honeynet configuration links honeypots to the actual network. There appears to have been a typo or miscommunication. The term "Honeybrid" here does not relate to a particular kind of honeypot, but rather to a gateway or bridge. Honeybrid regulates and reroutes traffic between the honeynet and the real network using alerts from an IDS/IPS system. It is not a particular type of honeypot; rather, it is a component of the system that controls traffic routing.

IDS/IPS alerts are used by Honeybrid to determine which traffic should be sent to the honeynet. The traffic travels to the server via a hybrid gateway if no warning is raised. From the IDS/IPS alerts, the Decision Engine selects the most crucial traffic and forwards it to the honeynet [30]. The Honeybrid gateway is composed of these two components.

Different kinds of honeypots, including High Interaction (HIH), Low Interaction (LIH), and Medium Interaction (MIH), can be found in a honeynet. The honeynet aids in verifying the attacks that the IDS/IPS have detected. It can be moved to the cloud for improved management and security because it is a centralized system. For improved scalability and reduced resource use, a virtual honeynet is advised. TPOT CE, an open-source framework for building and maintaining cybersecurity honeypots, is used to set up the honeynet in our experiment. Traffic is first examined by IDS/IPS at a gateway before it reaches the server. The communication is routed to a honeypot inside the honeynet if an assault is identified. It is transmitted to the server if no attack is found. Following Figure 14 shows the Honeypot Architecture Installation Sequence.

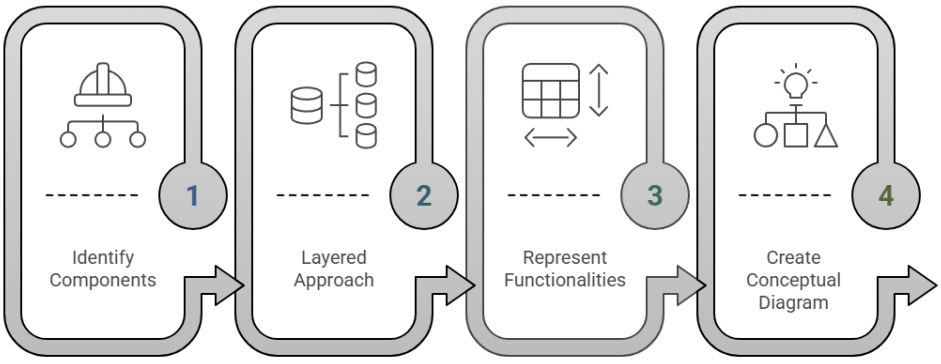



**Figure 14: Honeypot Architecture Installation Sequence**

Alibaba Cloud is the configuration for the honeypot system. Low-interaction [65] tools that are operating on the virtual machine. There, the honeypot system will be combined with a module that packs force

learning [31]. Network traffic is recorded using the public IP address. Between September 1st, 2022, and October 31st, 2022, the deployed honeypot was operational. The most recent versions of Twisted dependencies, Python 3.8, and Cowrie (1.5.1) are used in this study.
The honeypot system allowed connections via SSH, MySQL, and HTTP on a number of ports (2222,3306, and 80). Examples of ports that are permitted via the honeypot system are HTTP connections, MySQL, and SSH. More traffic was handled by SSH than by any other network service protocol. SSH traffic (successful SSH E. Data Preprocessing connections) includes SSH breaches, brute-force attacks, and scan assaults.

**11. Evolution of Honeypot**

Honeypots are a crucial part of current cybersecurity plans since they improve defensive measures and offer unmatched insights into harmful activity [15]. Originally, honeypots were straightforward devices used to track and record network activity, which assisted in identifying attackers looking for weaknesses. APTs were difficult to detect using early honeypots, which provided little interaction and were useful for learning the fundamental behaviors of attackers. Deeper insights into attacker strategies, techniques, and processes were made possible by the emergence of high-interaction honeypots as cyber threats increased [49].
They are now much more realistic and less detectable by recent developments in honeypot design. In order to scale deployments and improve efficacy, modern approaches like virtual honeypots and honeypot farms have been created. Virtual honeypots generate several separate honeypot instances on a single physical machine by leveraging virtualization technology[15].
In contrast to other security solutions, honeypots are active defence technologies that use a few hosts and network services set up as lure to identify and examine attack patterns[17]. Honeypot farms are groups of honeypots intended to provide thorough coverage and detection capabilities throughout a network by covering a wide variety of attack vectors and services [15].
By simulating different operating systems, apps, and network configurations, virtual honeypots offer a flexible tool for identifying and evaluating diverse attack types [15]. Originally, honeypots were used to research hackers. They were initially straightforward and employed for research. Honeypots were employed to identify network vulnerabilities as cyber threats become more severe. They were then employed in actual networks to identify intrusions and discover hacker techniques. By sharing the data with specialists, the honeypot data contributed to increased security. Cloud technologies made using honeypots less expensive and easier [30]. A layered defence strategy is provided by honeypot farms, which are made up of several honeypots that cooperate to cover a variety of attack surfaces. These honeypot farms can be dynamically scaled and managed with orchestration tools, enabling automated deployment and monitoring of honeypot environments. They can include both low-interaction and high-interaction honeypots [15].
Now digital avatars of real-world people and items live in a virtual environment called the Metaverse. According to the Metaverse-Honeypot (MV-Honeypot) idea, the Metaverse infrastructure and its constituent parts include vital information that draws in attackers looking to take advantage of system flaws. As a result, there are serious hazards associated with network connectivity, data management, access control, authentication, and exchanging data with outside parties in the Metaverse. Since both Avatars and the Metaverse rely on software, a cyberattack that targets the data flow of the SARANG model could seriously risk user privacy and interfere with Avatars' ability to function in the Metaverse. To safeguard Avatar security from cyberattacks in the Metaverse, it is essential to create strong applications and defenses [41]. As we know, honeypot architectures have developed, traditional central models have frequently shown flaws that risk the accuracy and dependability of the data gathered. We offer an innovative way to overcome these drawbacks by incorporating blockchain technology. Blockchain, which was first created for cryptocurrencies like Bitcoin, offers a decentralized, impenetrable structure that greatly improves the security and robustness of honeypot networks. Every important protocol required for Bitcoin's vital operations, including as blockchain, consensus, key management, and networking protocols, should be covered by security solutions [61].

With the help of blockchain's fundamental ideas of immutability, decentralization, and transparency, we want to develop a more resilient, dispersed honeypot system that can withstand sophisticated cyberattacks[43]. Each honeypot event is recorded as a block in the system's blockchain-based architecture, which guarantees the data is safe and unchangeable. To automatically take steps, such as blocking IP addresses when harmful behavior is identified, it also makes use of smart contracts. Furthermore, the technology facilitates peers' real-time exchange of threat intelligence, enhancing security teamwork. Through the integration of smart contracts, blockchain, and threat intelligence sharing, a strategy gives businesses a more robust and efficient defence against online attacks [59].

A cloud-based system called Honeypot-as-a-Service (HaaS) uses Kubernetes Clusters to automate the deployment, upkeep, and removal of containerized honeypots. Through a dashboard, users may set up honeypots without having to worry about maintaining the infrastructure. It was created in the cybersecurity department to make network defence honeypot deployment easier. Although its performance and availability satisfy cloud service requirements, more work is required to improve functionality [48].

Last year, there a drastic changes happens BHICS is an innovative, dynamic honeypot conversion system with blockchain-enabled logging and adaptive resource allocation that improves IoT network security. Our analysis shows that dynamic honeypot conversion greatly increases resource efficiency while offering security on par with dedicated honeypot deployments. When compared to unprotected networks, BHICS reduces node compromise rates by more than 50% and achieves attack prevention rates of over 76%. Notably, even with high attack loads and across different network sizes (from 100 to 1,000 nodes), blockchain transaction speeds typically stay low, averaging about 15 milliseconds, guaranteeing no influence on performance. In bigger deployments, the system's efficacy increases with network size, offering improved protection with a compromise rate as low as 2.9% at 1,000 nodes. It continues to function consistently even when attack traffic increases by 10–50% [51]. Initially, blockchain-based verification confirms the legitimacy of IoT devices before data transfer by using the Deoxys verification Algorithm (DAA).

**12. Integration of Honeypot in ChatGPT**

There were multiple procedures needed in integrating a honeypot system with ChatGPT to guarantee effortless interaction between the two platforms. The first thing we did was link ChatGPT to the honeypot. Every contact the honeypot had with the attackers was routed to ChatGPT for examination. We used an API, a mechanism that enables communication across several software systems, to establish this link. Here, we connected ChatGPT and the honeypot using OpenAI's API [34].Once the connection was established, we configured ChatGPT to react to attackers in a way that prompted them to divulge their strategies and intentions. To do this, we used human-attacker dialogues to train the model. This made it easier for the model to produce relevant and compelling answers. Because ChatGPT was made to respond in a way that seemed human, it was more difficult for attackers to identify that they were dealing with a honeypot. By using this tactic, we were able to learn important details about the attackers without raising any red flags.ChatGPT and the honeypot system work together to provide a robust defence against online attacks. While ChatGPT interacts with attackers to gain a better understanding of their strategies and objectives, the honeypot draws them in and records their activities. By working together, organizations may better understand how attackers work and develop defenses against cyberattacks. Following Figure15 shows the Integration of Honeypot Architecture with Chatgpt.

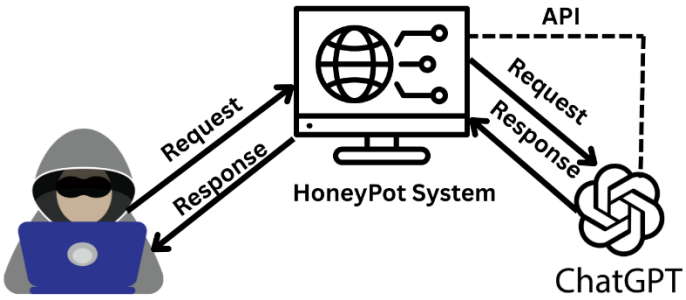



**Figure 15: Integration of Honeypot Architecture with Chatgpt**

### 13. Challenges of Honeypot

The design of a honeynet is to provide a highly controlled network in which all activity can be contained and recorded. Data control and data capture are two essential components of honeynet design. Data control describes how an action is kept hidden from the attacker inside the honeynet. Logging every action without the attacker's knowledge is known as data capture. Data acquisition is never more important than data control [30].

The possibility of being found by attackers is one of the main difficulties in establishing honeypots. Professional adversaries may frequently detect honeypots using a variety of methods, including searching for odd responses, keeping an eye on network traffic patterns, or employing tools built to find honeypot signatures. Attackers may completely avoid a honeypot after they have been located, which lessens its usefulness. Numerous sources, including Joshi (2011), Jones, and Martinez (2018), have emphasized the difficulties in deploying honeypots, including false positives and excessive resource use. Successfully incorporating honeypots into network security requires a thorough awareness of and attention to these problems [36].Detected honeypots may occasionally even be used as ruses by attackers to trick security personnel into attacking other parts of the network [15].

### 13.1 Needed Resources

Sufficient gear, network bandwidth, and knowledgeable personnel are necessary for an efficient honeypot implementation. Maintaining security also requires frequent software and system updates for honeypots.

### 13.2 Misleading Negative Threats

Because they could result in security concerns going unnoticed, false negatives can be a serious problem. It's crucial to update systems frequently, keep a careful eye on them, work with other security measures, and utilize extra security controls in addition to honeypots to lower the possibility of false negatives.

### 13.3 Legal Consequences

Privacy, adhering to privacy regulations, and handling the data gathered via honeypots are all legal issues. It's critical to adhere to privacy laws, use data judiciously, and establish explicit protocols for collaborating with law enforcement [36].

Only companies with large cybersecurity resources and experience can afford to deploy and maintain high-interaction honeypots due to their complexity and resource requirements. The use of honeypots brings up several moral and legal concerns. Entrapment, where there is a thin line between inviting an attacker to interact with a honeypot and illegally attracting them into committing a crime, is one of the primary ethical issues. In addition to allowing organizations to deploy honeypots across remote environments, which increases coverage and detection capabilities, cloud-based honeypots are also easily deployed and managed, providing flexibility in responding to shifting threat landscapes [15].

### 14. Types of Honeypot Tools

With their wide range of features for identifying and evaluating possible threats, honeypot technologies are essential in the field of cybersecurity. These solutions, which are uniquely suited to particular security requirements and applications, can be broadly divided into three categories: open-source honeypots, commercial honeypots, and traditional security tools.

An outline of these categories and their corresponding tools is given below:

### 14.1 Traditional Security Tools

Even though many security tools, such as intrusion detection systems (IDSs), firewalls, intrusion prevention systems (IPSec), and threat monitoring systems, are made to identify and stop attacks, they have several drawbacks. They have trouble identifying unknown and zero-day threats. They are not particularly good at closely examining attacker behavior.

#### 14.1.2 Tools for Commercial Honeypots

##### 14.1.2.1 Netbait

This tool uses honeypot technology to protect corporate networks. By providing misleading information about the system, Netbait draws in attackers and traps them while they discover new ways to breach the system. It can be used as a research honeypot to find new attack techniques or as a production honeypot to safeguard corporate assets.

##### 14.1.2.2 Mantrap

Resource Technologies created this high-interaction honeypot, which uses the "cage" concept to keep intruders out. To stop attackers from escaping or focusing on the host system, it generates several virtual cages that are derived from a fully functional operating system. This method provides a reliable and adaptable way to secure real-world settings.

##### 14.1.2.3 Spectre

To attract attackers, the intrusion detection system Spectre simulates a weak system. It offers several services as traps for attackers, including FTP, SMTP, POP3, HTTP, and TELNET. Spectre records every action they do and alerts the administrator, allowing for thorough surveillance and analysis without disclosing that they are communicating with a decoy system.

##### 14.1.2.4 KFsensor

KFsensor is an intrusion detection system that uses honeypots and is made especially for Windows. It has a signature-based engine, remote management capabilities, and banner technology for port monitoring, among other innovative and unique characteristics. These characteristics improve its ability to identify and neutralize possible risks.

### 14.2 Open Source Tools for Honeypots

#### 14.2.1 Friendly Back Officer

This straightforward, open-source honeypot utility is well-known for being simple to set up, deploy, and maintain. TCP, HTTP, Black Orifice, and seven other services are supported. Windows 98 and Windows 95 are among the Windows operating systems that the tool works with.

#### 14.2.2 Switch and Bait

This readily available honeypot program is intended to draw in attackers and pinpoint popular techniques for system compromise. It is available in several versions, each with new features for improved functionality.

#### 14.2.3 Labrea Tarpit

Labrea Tarpit logs attacker activity and notifies security professionals by acting as a decoy system inside a network. It is a low-interaction honeypot that uses an emulated operating system and is made to successfully entice hackers.

#### 14.2.4 Honeyd

Honeyd is a free, low-interaction honeypot made to function in simulated environments by Neils Provos of the University of Michigan. It gathers comprehensive data about attackers, including their IP addresses, tools, ports they target, and attack techniques.

#### 14.2.5 DTK (Deception Toolkit)

The Deception Toolkit, developed by Fred Cohen, may mimic a variety of system services. Additionally, it can pose as several hosts, offering features akin to those of a honey net for improved surveillance and deception [45].

## 15. Future Techniques

More sophisticated deception strategies will be used by future honeypots to increase their realism and lower the chance of discovery. This involves establishing complex and realistic network environments, copying real user interactions with deep-fake technology, and copying authentic user

activity. These methods seek to increase the accuracy of honeypots in attracting and understanding attackers by making them indistinguishable from authentic systems. In addition to allowing organizations to deploy honeypots across remote environments, which increases coverage and detection capabilities, cloud-based honeypots are also easily deployed and managed, providing flexibility in responding to shifting threat landscapes [15].

**Table 4: Comparison of traditional security tools, commercial honeypots, and open-source honeypots by features, limitations, use cases, and examples.**

| Category | Features | Limitations | Use Cases | Examples |
|---|---|---|---|---|
| Traditional Security Tools | - Uses pre-established rules and signatures to identify known risks.<br>- Contains IPSec, IDSs, and firewalls.<br>- Offers monitoring in real time. | -Faced challenges from unknown and zero-day threats.<br>-The ability to analyze specific attacker behaviors is limited. | -The defence of perimeters. Preventing known attack points.<br>- Traffic observation. | Threat Monitoring Systems, Firewalls, IPSec, and IDSs |
| Commercial Honeypot | -Tailored to business settings.<br>- Integrates study of attacker behavior with deception.<br>-Research and production options are available. | -Expensive in contrast to open-source substitutes.<br>-Technical know-how is necessary for an efficient setup. | -High-security business settings.<br>-The study of sophisticated attack methods. | Mantrap, Spectre, KFsensor, and Netbait |
| Netbait | -False information is used to entice attackers.<br>- Captures attackers while they are learning their techniques.<br>-It functions in either production or research mode. | -Restricted to settings with business networks; might not be able to manage situations with a lot of engagement. | -Investigating new attack techniques.<br>- Preserving company property. | Netbait |
| Mantrap | -A honeypot with high interaction.<br>-It generates several virtual cages based on an operating system.<br>-Prevents the host from being the target of an attack. | -A high-interaction honeypot.<br>-Depending on the operating system, it creates multiple virtual cages.<br>-Stops an attack from being directed against the host. | -Protecting environments of great value.<br>-In-depth examination of the tactics used by attackers. | Mantrap |
| Spectre | - Uses traps to imitate weak systems (FTP, SMTP, POP3, HTTP, TELNET).<br>- Documents the | -Only low-complexity assaults are allowed.<br>-It is susceptible to being recognized as | - Keep an eye on the actions of attackers.<br>-Carrying out thorough surveillance. | Spectre |

| | | | | |
|---|---|---|---|---|
| | activity of the attacker.<br>-Notifies administrators. | a honeypot by attackers. | | |
| KFsensor | -It was made for Windows.<br>-It has features like port monitoring, remote management, and detection based on signatures.<br>-Banner technology is used. | -It is limited to Windows OS.<br>-It could not be scaled for big networks. | - Tracking systems that run on Windows.<br>-Identifying and eliminating dangers. | KFsensor |
| Open Source Honeypot Tools | -It is freely accessible.<br>-Deployment and customization are simple.<br>-It works well for research projects or smaller-scale initiatives. | -Enterprise-grade support is frequently absent.<br>-It may require a lot of conFigureuration work. | -Research environments.<br>-Small-scale or academic deployments. | Labrea Tarpit, Honeyd, DTK, Switch and Bait, and a friendly back officer |
| Friendly Back Officer | -Easy to set up and maintain.<br>-TCP, HTTP, Black Orifice, and more protocols are supported.<br>-It is compatible with previous versions of Windows OS. | -Compatibility with contemporary operating systems is limited.<br>- Does not possess sophisticated deception skills. | -Minimal-scale settings.<br>-Simple honeypot operations. | Friendly Back Officer |
| Switch and Bait | - Concentrates on recognizing typical attack methods.<br>-Frequent upgrades to enhance functionality. | -Scalability is limited.<br>-It might not offer a thorough behavioral examination. | -Compiling data on frequent attacks.<br>-Minimal settings for research. | Bait and Switch |
| Labrea Tarpit | -A honeypot with little interaction.<br>- Records the actions of the attacker.<br>-Emulated operating systems are used to draw hackers. | -Low-interaction capabilities are the limit.<br>-Advanced attack scenarios are unsuitable. | -Decoys on the network.<br>Analyzing and recording basic attacks. | Tarpit Labrea |

| | | | | |
|---|---|---|---|---|
| Honeyd | -It can simulate a variety of virtual systems.<br>-Compiles comprehensive information on the attackers.<br>-Customizable and open-source. | -Design with little interaction.<br>-It is not appropriate for high-security settings. | -Investigate the tools and techniques used by attackers.<br>-Academic settings. | Honeyd was created by ( University of Michigan's Neils Provos ) |
| DTK (Deception Toolkit) | -It imitates several system functions.<br>-Capable of simulating several hosts at once.<br>-It works similarly to a honey net. | -Needs technological know-how.<br>-Limited enterprise-use scalability. | -Advanced techniques for deception.<br>- Monitoring of several systems. | Fred Cohen created the Deception Toolkit (DTK). |
| Future Techniques | -Realistic network simulations.<br>-Deepfake technology for how users behave.<br>-Honeypots in the cloud for flexibility. | -It is still being developed.<br>Complexity may result in performance overheads. | -Improving the realism of honeypots.<br>- Adjusting to changing threat environments. | Advanced AI-driven deception strategies and cloud-based honeypots |

## 16. Proposed Methodology

Several important gaps still need to be addressed despite the valuable contributions of existing research in cybersecurity, holes that need to be filled, particularly in protecting against new threats like ransomware attacks, zero-day vulnerabilities, and APTs. Scalability, adaptability, and real-time threat detection while protecting data privacy are issues that traditional honeypot systems must deal with. We suggest the Federated Quantum-Aided Honeypot Ecosystem (FQAHE) as a solution to these problems. Federated Learning (FL), Quantum Computing, Blockchain, and Software-Defined Networking (SDN) are all combined in this innovative framework to produce a honeypot system that is efficient, scalable, and flexible.

### 16.1 Architecture of the System

A multi-layered architecture serves as the foundation for the suggested FQAHE framework, ensuring thorough threat detection and response. The following layers define the design of the system:

**Layer 1:** Integration of Federated Learning
**Goal:** Preserve data privacy while facilitating decentralized learning. Every honeypot node, such as those placed in various geographic regions, gathers attacker data on its own and uses it to build local machine learning models. Without exchanging raw data, these models are routinely combined into a global model using FL approaches. This layer correlates distributed data to detect zero-day exploits and complex attack patterns, allowing for real-time anomaly detection and behavioral analytics.

**Layer 2:** Quantum Randomization Layer (QRL)
**Goal:** The goal is to make honeypot behavior unpredictable, simulate realistic system behaviors by generating random data streams using Quantum Random Number Generators (QRNGs). Because of the randomization, honeypot actions are dynamically adjusted, making it difficult for attackers to examine and get around the system.

**Layer 3:** Immutable Logging Using Blockchain Technology

**Goal:** Make sure that attacker activity is securely and impenetrably logged. A decentralized blockchain ledger records every interaction with the honeypot, offering forensic analysts unchangeable and verifiable recordings. Smart contracts are used to automate threat responses, including traffic redirection or IP blocking.

**Layer 4:** Honeypot Architecture with Multiple Layers
**Goal:** Gather comprehensive data and maximize resource utilization. Low-interaction honeypots: Find general attack trends while using the fewest resources possible. High-interaction honeypots: Get in-depth information from attackers. Hybrid honeypots: To increase effectiveness, combine the advantages of high- and low-interaction honeypots.

**Layer 5:** Advanced Deception Mechanisms (ADM)
**Goal:** To trick attackers, imitate real-world systems. Uses high-interaction honeypots and quantum simulations to produce erratic settings that mimic actual systems, increasing the realism of the honeypot environment by simulating real user interactions using Deepfake technology.

**Layer 6:** Software-Defined Networking (SDN)
**Goal:** Effectively isolate threats while causing the fewest possible interruptions. Protects normal network traffic by dynamically rerouting suspicious traffic to the proper honeypot levels. Guarantees a smooth integration with the current network architecture for improved threat mitigation and isolation. Following Figure 16 shows the FQAHE Framework Architecture.

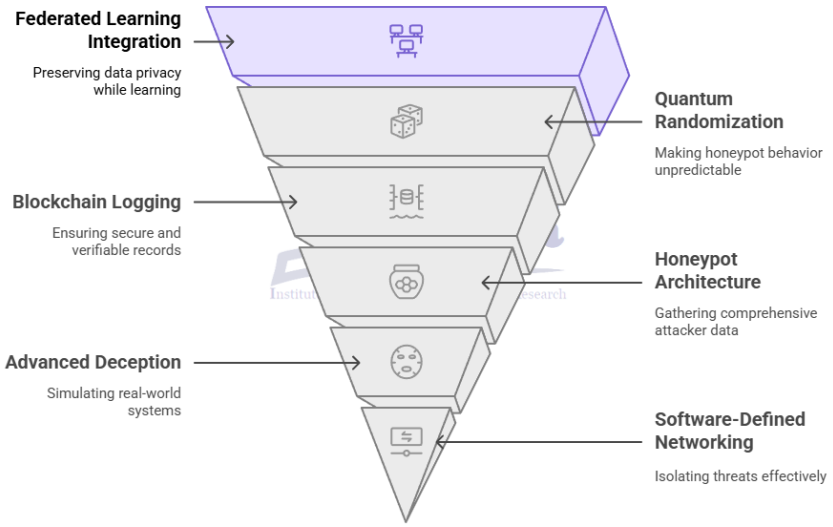



**Figure 16: FQAHE Framework Architecture**

### 16.3 Example

A honeypot—a decoy system intended to entice and trap attackers—is placed at each of the three locations where a corporation operates: the US, the UK, and Pakistan.

**US honeypot:** This honeypot records the hacker's tools and methods when they try a ransomware attack.

**Pakistan honeypot:** An SQL injection attack is launched by another hacker, and the attacker's inputs are recorded by this honeypot.

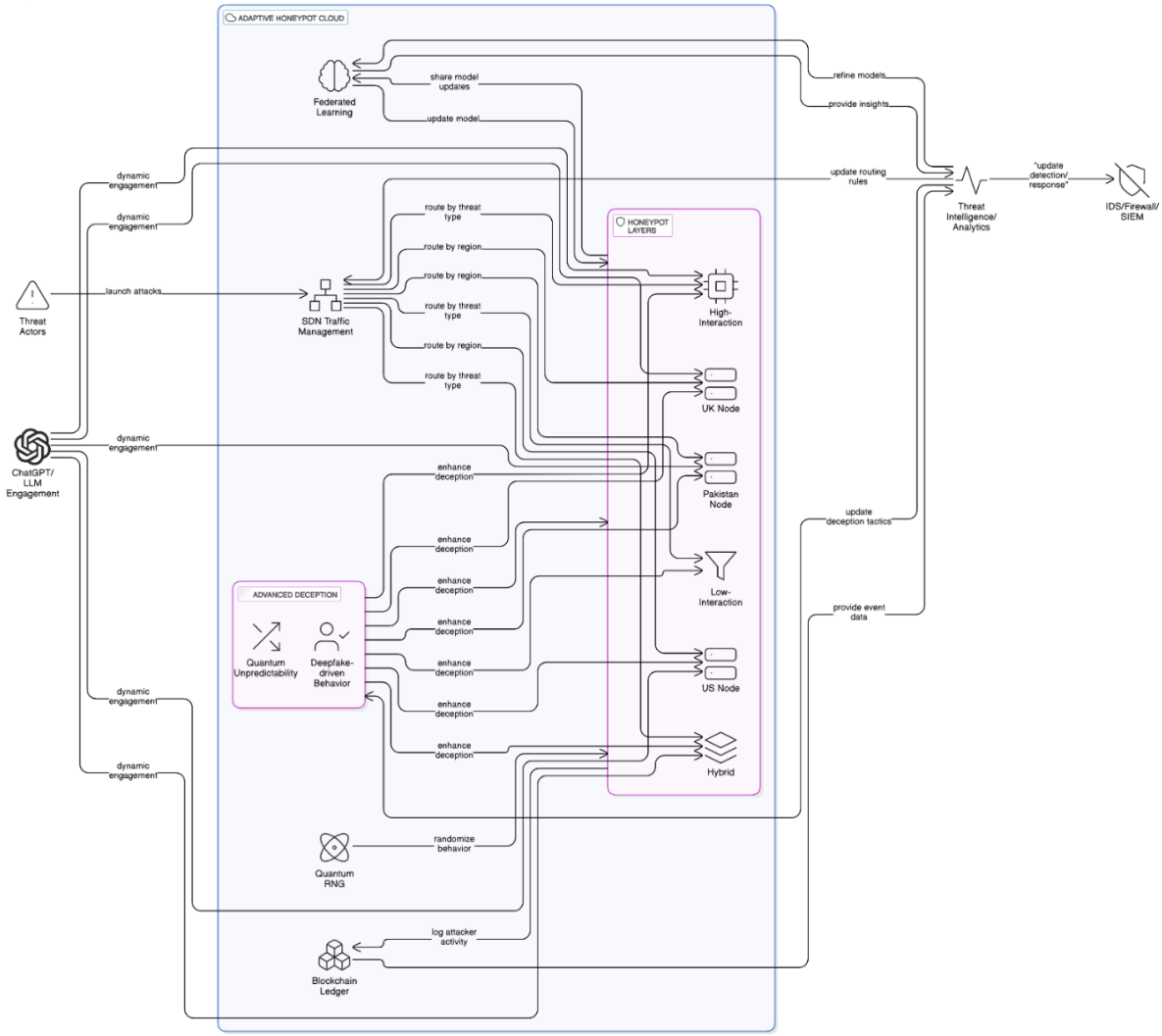
**UK honeypot:** This honeypot finds an attack when a hacker performs port scanning, which looks for network weaknesses.

By analyzing their data, these three honeypots build a common system using federated learning. This implies that without sharing any private information, the honeypots in the US, India, and the UK share summaries of their findings with a central model.

Suspicious traffic is dynamically redirected to the honeypots by the SDN system, protecting the actual systems. All hacker activity is securely recorded by the blockchain technology, which is impenetrable. Every new attack makes this honeypot system wiser, allowing

it to understand the hackers' methods and better prepare for potential threats.

## 17. Diagrammatic Model




## 18. Conclusion

To identify and address changing cyberthreats, this study effectively deployed and analyzed adaptive honeypot architectures along with cutting-edge technologies including artificial intelligence (AI), reinforcement learning, and blockchain. The study looked at several kinds of honeypots, deception techniques, and how they were used in actual settings, such as cloud and Internet of Things settings. Improved threat detection capabilities were shown by real-world applications including web portal honeypots and Kubernetes containerized deployments. New insights on attacker interactions were obtained through the integration with ChatGPT. In the end, this study established the groundwork for upcoming advancements in proactive threat intelligence and mitigation in addition to achieving its goal of putting out a dynamic, scalable cybersecurity defense system.

**References**

[1] M. Gupta, M. Abdelsalam, S. Khorsandroo and S. Mittal, "Security and Privacy in Smart Farming: Challenges and Opportunities," in *IEEE Access*, vol. 8, pp. 34564-34584, 2020, doi: 10.1109/ACCESS.2020.2975142.

[2]Mohamed Ali Setitra, Mingyu Fan, Ilyas Benkhaddra, Zine El Abidine Bensalem,DoS/DDoS attacks in Software Defined Networks: Current situation, challenges and future directions,Computer Communications, https://doi.org/10.1016/j.comcom.2024.04.035.

[3] Amir Javadpour, Forough Ja'fari, Tarik Taleb, Mohammad Shojafar, Chafika Benzaïd, A comprehensive survey on cyber deception techniques to improve honeypot performance, Computers & Security, ISSN 0167-4048,https://doi.org/10.1016/j.cose.2024.103792.

[4] Xingsheng Qin, Frank Jiang, Mingcan Cen, Robin Doss,Hybrid cyber defense strategies using Honey-X: A survey,Computer Networks,Volume 230,2023,109776,ISSN 1389-1286, ttps://doi.org/10.1016/j.comnet.2023.109776.

[5]Ahmad Z, Shahid Khan A, Wai Shiang C, Abdullah J, Ahmad F. Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Trans Emerging Tel Tech*. 2021; 32:e4150. https://doi.org/10.1002/ett.4150.

[6]Kolehmainen, A. (2024). *Edge Infrastructure Testbeds as Tools for Understanding Security Management in IoT*. (Tampere University Dissertations - Tampereen yliopiston väitöskirjat; Vol. 1141). Tampere University. https://urn.fi/URN:ISBN:978-952-03-3709-4

[7] Beltrán López, P., Nespoli, P., & Gil Pérez, M. (2024). Cyber deception reactive: TCP stealth redirection to on-demand honeypots. arXiv.

[8] Balamurugan, Merlin. (2024). AI-enhanced Honeypots for Zero-Day Exploit Detection and Mitigation. International Journal For Multidisciplinary Research. 6.

[9] **Aggrey, R., Adjirachor, E., Adjei, B. A., Dsane, N. A., & Afoduo, K. O. (2024).** Predicting future cyber threats: Analyzing trends and predicting future cybersecurity challenges. *International Journal for Multidisciplinary Research*, *6*(6).

[10] N. Eliot, D. Kendall and M. Brockway, "A Flexible Laboratory Environment Supporting Honeypot Deployment for Teaching Real-World Cybersecurity Skills," in *IEEE Access*, vol. 6, pp. 34884-34895, 2018, doi: 10.1109/ACCESS.2018.2850839.

[11] Evaluation and Comparison of the Use of Reinforcement Learning Algorithms on SSH Honeypot. (2024). *Teknika*, *13*(1), 77-85. https://doi.org/10.34148/teknika.v13i1.763.

[12] Morić, Z., Dakić, V., & Regvart, D. (2025). Advancing Cybersecurity with Honeypots and Deception Strategies. *Informatics*, *12*(1), 14. https://doi.org/10.3390/informatics12010014.

[13] Chiran, Jananga. (2024). Comprehensive Literature Review on Honey Pot design. 10.5281/zenodo.14677746.

[14] Tan, R.R., Eng, S., How, K.C., Zhu, Y., Jyh, P.W.H. (2023). Honeypot for Cybersecurity Threat Intelligence. In: Guo, H., *et al.* IRC-SET 2022. Springer, Singapore.

[15] Pathiraja, P. M. C. H., & Hiruni, C. (2024). Decoding cyber threats: A comprehensive review of honeypot designs. *International Journal of Computer Applications*, *182*(35), 1–7.

[16] Li Li, Hua Sun and Zhenyu Zhang, "The research and design of honeypot system applied in the LAN security," *2011 IEEE 2nd International Conference on Software Engineering and Service Science*, Beijing, 2011, pp. 360-363, doi: 10.1109/ICSESS.2011.5982237.

[17] W. Tian et al., "Prospect Theoretic Study of Honeypot Defense Against Advanced Persistent Threats in Power Grid," in IEEE Access, vol. 8, pp. 64075-64085, 2020.

[18] J. You, S. Lv, Y. Sun, H. Wen and L. Sun, "HoneyVP: A Cost-Effective Hybrid Honeypot Architecture for Industrial Control Systems," *ICC 2021 - IEEE International Conference on Communications*, Montreal, QC, Canada, 2021, pp. 1-6, doi: 10.1109/ICC42927.2021.9500567.

[19] Bauraitė, A., Brilingaitė, A. & Bukauskas, L. (2024). Designing Trainee Performance Assessment System for Hands-On Exercises. In B. Marcinkowski, A. Przybylek, A. Jarzębowicz, N. Iivari, E. Insfran, M. Lang, H. Linger, & C. Schneider (Eds.), *Harnessing Opportunities: Reshaping ISD in the post-COVID-19 and Generative AI Era (ISD2024 Proceedings)*. Gdańsk, Poland: University of Gdańsk. ISBN: 978-83-972632-0-8.

[20] Lanka, P., Gupta, K., & Varol, C. (2024). Intelligent Threat Detection—AI-Driven Analysis of Honeypot Data to Counter Cyber Threats. *Electronics*, *13*(13), 2465 https://doi.org/10.3390/electronics13132465 .

[21] J. Franco, A. Aris, B. Canberk and A. S. Uluagac, "A Survey of Honeypots and Honeynets for Internet of Things, Industrial Internet of Things, and Cyber-Physical Systems," in *IEEE Communications Surveys & Tutorials*, vol. 23, no. 4, pp. 2351-2383, Fourthquarter 2021, doi: 10.1109/COMST.2021.3106669.

[22] J. Z. Guizhou and Z. Liu, "New honeypot system and its application in security of employment network," *2012 IEEE Symposium on Robotics and Applications (ISRA)*, Kuala Lumpur, Malaysia, 2012, pp. 627-629, doi: 10.1109/ISRA.2012.6219267.

[23] N. Bhagat and B. Arora, "Intrusion Detection Using Honeypots," *2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, Solan, India, 2018, pp. 412-417, doi: 10.1109/PDGC.2018.8745761.

[24] Kumar, L. K. S. . (2021). Intrusion Detection System through ID3-PCA-BP. *The Journal of Contemporary Issues in Business and Government*, *27*(6), 1802–1821. Retrieved from https://cibgp.com/au/index.php/1323-6903/article/view/2274.

[25] Morić, Z., Dakić, V., & Regvart, D. (2025). Advancing Cybersecurity with Honeypots and Deception Strategies. *Informatics*, *12*(1), 14. https://doi.org/10.3390/informatics12010014.

[26] Xingyuan Yang & Jie Yuan & Hao Yang & Ya Kong & Hao Zhang & Jinyu Zhao, 2023. "A Highly Interactive Honeypot-Based Approach to Network Threat Management," Future Internet, MDPI, vol. 15(4), pages 1-31, March.

[27] Zou, J., Sun, Z., Ku, C., Li, X., & Dahbura, A. (2024). WiP: Developing high-interaction honeypots to capture and analyze region-specific bot behaviors. In *Proceedings of the Hot Topics in the Science of Security Symposium (HotSoS 2024)* (pp. 1–2). Johns Hopkins University.

[28] M. A. Lihet and V. Dadarlat, "How to build a honeypot System in the cloud," *2015 14th RoEduNet International Conference - Networking in Education and Research (RoEduNet NER)*, Craiova, Romania, 2015, pp. 190-194, doi: 10.1109/RoEduNet.2015.7311992.

[29] Ahmed, Y., Beyioku, K., Yousefi, M.: Securing smart cities through machine learning: a honeypot-driven approach to attack detection in Internet of Things ecosystems. *IET Smart Cities*. 6(3), 180–198 (2024). https://doi.org/10.1049/smc2.12084.

[30] Sharma, C. R. (2024). *Enhancing false positive detection in IDS/IPS using honeypots: A case study with CSE-CIC-2018 dataset* (Master's thesis, International Institute of Information Technology Hyderabad).

[31] M. A. Kristyanto, H. Studiawan and B. A. Pratomo, "Evaluation of Reinforcement Learning Algorithm on SSH Honeypot," *2022 6th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, Yogyakarta, Indonesia, 2022, pp. 346-350, doi: 10.1109/ICITISEE57756.2022.10057816.

[32] Chandrashekar, K., & Jangampet, V. D. (2024). Honeypots as a proactive defense: A comparative analysis with traditional anomaly detection in modern cybersecurity. *International Journal of Computer Engineering & Technology*, *10*(5), Article 21 https://doi.org/10.34218/ijcet_10_05_021.

[33] Patel, P., Dalvi, A., & Sidddavatam, I. (2022). Exploiting honeypot for cryptojacking: The other side of the story of honeypot deployment. In *Proceedings of the 2022 6th International Conference on Computing, Communication, Control and Automation (ICCUBEA)* (pp. 1–5). IEEE. https://doi.org/10.1109/ICCUBEA54992.2022.10010904.

[34] U. Raut, A. Nagarkar, C. Talnikar, M. Mokashi and R. Sharma, "Engaging Attackers with a Highly Interactive Honeypot System Using ChatGPT," *2023 7th International Conference On Computing, Communication, Control And Automation (ICCUBEA)*, Pune, India, 2023, pp. 1-5, doi: 10.1109/ICCUBEA58933.2023.10392228.

[35] C. Moore, "Detecting Ransomware with Honeypot Techniques," *2016 Cybersecurity and Cyberforensics Conference (CCC)*, Amman, Jordan, 2016, pp. 77-81, doi: 10.1109/CCC.2016.14.

[36] Kërnaja, A., Shtjefni, A., Hakrama, K., & Keçi, H. (2024). The use of photovoltaic technology in Albania. *Ingenious*, *4*(1), 7–20. European University of Tirana.

[37] S. Chamotra, J. S. Bhatia, R. Kamal and A. K. Ramani, "Deployment of a low interaction honeypot in an organizational private network," *2011 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC)*, Udaipur, India, 2011, pp. 130-135, doi: 10.1109/ETNCC.2011.5958501.

[38] Tan, R.R., Eng, S., How, K.C., Zhu, Y., Jyh, P.W.H. (2023). Honeypot for Cybersecurity Threat Intelligence. In: Guo, H., *et al.* IRC-SET 2022. Springer, Singapore. https://doi.org/10.1007/978-981-19-7222-5_44.

[39] Tulashvili Yurii, Kosheliuk Viktor. Orchestrating honeypot deployment in lightweight container platforms to improve security. International Science Journal of Engineering & Agriculture. Vol. 4, No. 1, 2025, pp. 1-13. doi: 10.46299/j.isjea.20250401.01.

[40] Kotwal, A. P. (2024). Leveraging big data analytics for enhanced cybersecurity: A comprehensive analysis of threat detection, incident response, and SIEM systems. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, *6*(6), Article CSEIT2410612414. https://doi.org/10.32628/CSEIT2410612414

[41]Alyas, Tahir, Alissa, Khalid, Alqahtani, Mohammed, Faiz, Tauqeer, Alsaif, Suleiman Ali, Tabassum, Nadia, Naqvi, Hafiz Hasan, Multi-Cloud Integration Security Framework Using Honeypots, *Mobile Information Systems*, 2022, 2600712, 13 pages, 2022. https://doi.org/10.1155/2022/2600712.

[42] Sarang, A.D., Alawami, M.A., Park, K. (2024). MV-Honeypot: Security Threat Analysis by Deploying Avatar as a Honeypot in COTS Metaverse Platforms. *Computer Modeling in Engineering & Sciences*, *141(1)*, 655–669. https://doi.org/10.32604/cmes.2024.053434.

[43] Nishad, N., & Singh, R. (2020). Honeypot deployment: A blockchain-based distributed approach. *International Research Journal of Modernization in Engineering, Technology and Science*, *2*(6), 599–603.

[44] J. Buzzio-Garcia, "Creation of a High-Interaction Honeypot System based-on Docker containers," *2021 Fifth World Conference on Smart Trends in Systems Security and Sustainability (WorldS4)*, London, United Kingdom, 2021, pp. 146-151, doi: 10.1109/WorldS451998.2021.9514022.

[45] B. Nagpal, N. Singh, N. Chauhan and P. Sharma, "CATCH: Comparison and analysis of tools covering honeypots," *2015 International Conference on Advances in Computer Engineering and Applications*, Ghaziabad, India, 2015, pp.

783-786, doi:10.1109/ICACEA.2015.7164809.

[46] Morozov, D. S., Vakaliuk, T. A., Yefimenko, A. A., Nikitchuk, T. M., & Kolomiiets, R. O. (2023). Honeypot and cyber deception as a tool for detecting cyber attacks on critical infrastructure. In *Proceedings of the International Workshop on Cybersecurity Advances (IWCyberSec 2023)* (Vol. 3374, pp. 55–60). CEUR-WS.org.

[47] M. M. Al-Hakbani and M. H. Dahshan, "Avoiding honeypot detection in peer-to-peer botnets," *2015 IEEE International Conference on Engineering and Technology (ICETECH)*, Coimbatore, India, 2015, pp. 1-7, doi: 10.1109/ICETECH.2015.7275017.

[48] Al-Hakbani, M. M., & Dahshan, M. H. (2015). Avoiding honeypot detection in peer-to-peer botnets. In *Proceedings of the 2015 IEEE International Conference on Engineering and Technology (ICETECH)* (pp. 1–7). IEEE. https://doi.org/10.1109/ICETECH.2015.7275017.

[49] Anıl Sezgin, Aytuğ Boyacı,DecoyPot: A large language model-driven web API honeypot for realistic attacker engagement,Computers & Security,ISSN 0167 4048, https://doi.org/10.1016/j.cose.2025.104458.

[50] A. L. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," in *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153-1176, Secondquarter 2016, doi: 10.1109/COMST.2015.2494502.

[51] Commey, Daniel and Nkoom, Matilda and Hounsinou, Sena and Crosby, Garth, Dynamic Honeypot Conversion for Enhanced IoT Security (January 01, 2025). Journal of Information Security and Applications, Forthcoming, Available at SSRN: https://ssrn.com/abstract=5079213 or http://dx.doi.org/10.2139/ssrn.5079213.

[52] Sam Maesschalck, Vasileios Giotsas, Benjamin Green, Nicholas Race, Don't get stung, cover your ICS in honey: How do honeypots fit within industrial control system security,Computers & Security,ISSN 0167-4048,https://doi.org/10.1016/j.cose.2021.102598.

[53] Otoum, Y., Asad, A., & Nayak, A. (2025). Blockchain meets adaptive honeypots: A trust-aware approach to next-gen IoT security. *arXiv preprint arXiv:2504.16226*. https://arxiv.org/abs/2504.16226.

[54] Alsabbagh, W., Urrego, D., & Langendörfer, P. (2025). *Smart traps for smart systems: Scalable honeynets for IIoT cybersecurity* [Preprint]. ResearchGate. https://doi.org/10.13140/RG.2.2.32342.15682.

[55] Song, Y. (2024). *Leveraging database honeypots to gather threat intelligence* (Master's thesis, Delft University of Technology). TU Delft Repository. https://doi.org/10.4233/uuid:ceb1acea-3301-4b51-8d23-915273bdaae9:contentReference[oaicite:5]{index=5}.

[56] Wang, Y., Gu, T., Teng, Y., Wang, Y., & Ma, X. (2025). HoneypotNet: Backdoor attacks against model extraction. *Proceedings of the AAAI Conference on Artificial Intelligence*, *39*(8), 8087–8095. https://doi.org/10.1609/aaai.v39i8.32872.

[57] L. Shi, Y. Li, T. Liu, J. Liu, B. Shan and H. Chen, "Dynamic Distributed Honeypot Based on Blockchain," in *IEEE Access*, vol. 7, pp. 72234-72246, 2019, doi: 10.1109/ACCESS.2019.2920239.

[58] Williams, J., Edwards, M., & Gardiner, J. (2025). Time-to-lie: Identifying industrial control system honeypots using the Internet Control Message Protocol. In *Proceedings of the 2025 IEEE 45th International Conference on Distributed Computing Systems (ICDCS)*. IEEE. https://doi.org/10.1109/ICDCS.2025.00055.

[59] Kakaraparthi, S., Immadisetty, D., & Maranco, M. (2024). Enhanced honeypot security for intrusion detection and prevention systems using blockchain. *World Journal of Advanced Research and Reviews*, *22*(1), 751–758. https://doi.org/10.30574/wjarr.2024.22.1.1065.

[60] Nisa, Z. (2023). *Honeypots: Concepts, Types, and Challenges*. ResearchGate. https://www.researchgate.net/publication/37

8164367_Honeypots_Concepts_Types_and_Challenges:contentReference[oaicite:5]{index=5}

[61] M. Conti, E. Sandeep Kumar, C. Lal and S. Ruj, "A Survey on Security and Privacy Issues of Bitcoin," in *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3416-3452, Fourthquarter 2018, doi: 10.1109/COMST.2018.2842460.

[62] Affinito, A., Zinno, S., Stanco, G., Botta, A., & Ventre, G. (2023). The evolution of Mirai botnet scans over a six-year period. Journal of Information Security and Applications, 79, 103629. https://doi.org/10.1016/j.jisa.2023.103629.

[63] Abe, S., Tanaka, Y., Uchida, Y., & Horata, S. (2018). Developing deception network system with traceback honeypot in ICS network. SICE Journal of Control, Measurement, and System Integration, 11(4), 372–379. https://doi.org/10.9746/jcmsi.11.372.

[64] Shu Z and Yan G. Ensuring Deception Consistency for FTP Services Hardened against Advanced Persistent Threats. Proceedings of the 5th ACM Workshop on Moving Target Defense. (69-79).https://doi.org/10.1145/3268966.3268971.

[65] *A honeypot architecture for detecting and analyzing unknown network attacks*. In **Kommunikation in Verteilten Systemen (KiVS), 14. ITG/GI-Fachtagung Kommunikation in Verteilten Systemen (KiVS 2005),** Kaiserslautern, Germany, February 28 – March 3, 2005. https://doi.org/10.1007/3-540-27301-8_20.