Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 2s (2024)

MHID: Malware Detection Using Hybrid Honeypot and Intrusion Detection System

Vinay Kumar Singh[1], Raj Sinha[2], Umang Garg[3]*, Rahul Kumar[4]*, Bhimasen Moharana[5], Parul Goyal[6]

[1] Professor, School of Computer Science and Engineering, Lovely Professional University, Punjab, India

[2]Assistant Professor, School of Computer Applications, Lovely Professional University, Punjab, India

[3, 4]Associate Professor, Department of Computer Science & Engineering, Amity School of Engineering and Technology, Amity University, Gwalior, Madhya Pradesh, India

[5]Assistant Professor, School of Computer Science& Engineering, Lovely Professional University, Punjab, India

[6]Professor, Computer Science & Engineering, M. M. Engineering College, Maharishi Markandeshwar Deemed to be University, Mullana, Ambala, Haryana, India

[1]vks.vinaykumarsingh@gmail.com, [2]rajsinha2310@gmail.com, [3]umangarg@gmail.com,

[4]rahulkumar1680@gmail.com, [5]bhimasen.moharana@gmail.com, [6]parul.goyal@mmumullana.org

*Corresponding Author

Article History:

Received: *20-03-2024*

Revised: *29-04-2024*

Accepted: *22-05-2024*

Abstract:

The latest wireless technology is developing smart phone technology and growing mobile cloud technology. Mobile cloud computing has a lot of potential benefits in the future, but it is also easy for hackers to gain complete control of many other users. Data privacy is crucial. While data security is designed to be secure, the most significant disadvantage for consumers is that once the computer is connected to the internet, an intruder can quickly steal data from the targeted target. To improve security, a hybrid intrusion detection system (HyInt) and honeypot networks have been added into the Mobile Cloud Environment, with the primary purpose of mitigating both unknown and known attacks.The study's execution offers a unique perspective on the algorithm's security and quality products that was not covered in previous studies. The research included extensive statistical analysis to demonstrate the consistency of the proposed algorithm. The implementation and evaluation findings demonstrate that there is lots of potential for further research in the cloud-based Intrusion Detection System. The developed technique can be used to effectively monitor network traffic in a high-security cloud environment designed for military and banking reasons.

Keywords: Malware, Intrusion Detection System, Machine Learning, Hybrid System, Honeypot Network.

1.INTRODUCTION

A honeypot can be used in network security to detect novel assaults that intrusion detection systems or network firewalls may be unable to detect using the traditional static protection rule method. When developing IDS and Firewalls, it is necessary to consider the corporate defensive measures for getting around the honeypot. Computer networks are vulnerable to a wide range of attacks that might make them unsecure or prohibit them from fulfilling their intended function. Intruders and attackers are becoming increasingly concerned about network security and impediments. Enterprises, organizations, and, most importantly, finance departments have an essay solution to embrace various hardware and software for network security providers such as firewalls, varieties of intrusion detectors, and [1] Virtual Private Networks to achieve better and improved security.These systems, on the other hand, operate constantly to keep personal information out of the hands of unscrupulous intruders and to alert users to new attacks as they occur.

Mobile Virtualization is the most advanced feature appearing in today's world, and its smartphone applications are increasing by the day. The number of mobile users is increasing all the time since it makes work easier and faster while also delivering the latest technology that is constantly improving and allowing users to access all their apps via the network from anywhere in the world. Cloud computing has a significant advantage in that it is incredibly versatile, and we may access data and share information anywhere in the world, even when we are not connected to the internet. It also provides cost-effectiveness in that use and maintenance are relatively inexpensive, as well as real-time data availability, which means that all user information is available in real time on our mobile device when connected to the network, allowing us to update and familiarize ourselves. Despite all the hype around the MCC, it has a significant shortcoming in terms of privacy and security, which contributes to trust difficulties for customers and enterprises. As the world evolves, so does the number of hackers.

Similarly, organizations are developing new security products and strategies for the cloud computing environment, where cloud computing services are supplied on a pay-as-you-go basis. How can the Hybrid Intrusion Detection System (HyINT) and Multi-Honeypot Network (MHN) work together to improve mobile cloud computing security?Honeypot networks are used to improve defence-in-depth defence and overall cloud security. Countermeasures require an examination of attack strategies, which is identified in the honeypot network. Many damaging threats, such as DDOS, XSS injection, and SQL injection, cannot be eliminated, but they can be mitigated. There are several methods for securing it against hackers, but IDS is the most important and extensively used method for identifying bad code in a network, and it is critical in protecting the cloud environment from attackers.

2.REVIEW OF LITERATURE

Pande et al. [1]identified numerous attacks on network infrastructures in the cloud network. Attacks on network availability, confidentiality, and integrity are among them. A distributed denial-of-service (DDoS) attack is a persistent attack that degrades network availability. The problem is approached using the Command and Control (C&C) method, which is utilized to carry out such an attack. To detect these attacks, many researchers have developed various ways based on machine learning techniques. In this paper, a DDoS attack was carried out with the ping of death approach and detected with the WEKA tool utilizing machine learning techniques. The NSL-KDD dataset was used in this experiment as a result.The normal and assault samples were classified using the random forest technique. The samples were accurately categorized in 99.76 percent of the time.

Arshi et al. [2] investigated the DDOS attacks occurred in the IoT network. These attacks are becoming more complicated, and the number of them is predicted to grow every day, making it difficult to detect and counteract them. The solution to the problem is to use a sophisticated IDS to discover and recognize abnormal internet traffic behavior. This article supports the approach by using the most recent dataset encompassing the most popular types of DDoS assaults, such as HTTP floods and SIDDoS. This study uses well-known grouping approaches like Naive Bayes, Multilayer Perception (MLP), and SVM, as well as decision trees. In the future, deep learning techniques will be used to undertake a comprehensive analysis of data sets encompassing the most recent types of attacks, such as HTTP flooding, SIDDoS, Smurf, and UDP flooding, on data obtained from the college network. Pei et al. [3] found that the harm caused by DDoS attacks is getting more serious.The issue is addressed by offering a machine learning-based DDoS attack detection system that includes two steps: feature extraction and model detection. A significant amount of DDoS attack traffic characteristics is recovered at the feature extraction stage by comparing data packages sorted according to rules. The experimental results show that the suggested machine learning-based DDoS assault detection strategy has a high detection rate for the most popular DDoS attack.

Sambangi et al. [4] highlighted detecting DDoS attacks as a machine learning classification challenge. The task of identifying DDoS attacks, which is pertinent to Cloud Computing, is extremely difficult due to the computational complexity that must be addressed.The goal is to examine the topic of DDoS attack detection in a cloud environment by using the most prominent CICIDS 2017 benchmark dataset and multiple regression analysis to create a machine learning model that predicts DDoS and Bot assaults based on a Friday afternoon traffic log file. The ensemble model achieved a prediction accuracy of 97.86 percent on the Friday morning dataset. The prediction accuracy for the Friday afternoon log file is 73.79 percent for 16 characteristics produced from an information gain-based feature selection and regression analysis-based machine learning model. Aytaç et al. [5] proposed implementing all required precautions to reduce losses by detecting assaults early. This study's approach to the topic is to examine the success rate of intrusion detection systems using various approaches.This study examined the CICDDoS2019 data set and compared DDOS assaults within it. Threat detection success rates were investigated using the following methods: Artificial Neural Networks (ANN), Support Vector Machine (SVM), Gaussian Naive Bayes, Multinomial Naive Bayes, Bernoulli Naive Bayes, Logistic Regression, K-nearest neighbor (KNN), Decision Tree (entropy-gini), and Random Forest. After analyzing five data sets with the highest accuracy rate in data trained with various algorithms, the Fwd Pkt Len Std, Fwd Seg Size Min, Bwd Pkt Len Std, Bwd Pkt Len Min, CWE Flag Count, Bwd Seg Size Avg, and Bwd Seg Size Min properties were discovered to be the most distinct values for determining DDOS attacks.

Fernando et al. [6] conducted a survey on the destructive nature of ransomware, the difficulties of reversing a ransomware infection, and the significance of identifying it before it infects a system, which are the primary motivations for this inquiry in this paper. Because machine learning and deep learning can detect zero-day threats, the solution to the problem is to research machine learning and deep learning approaches for ransomware detection. These methods can be used to build prediction models that can learn how ransomware behaves and then utilize that information to identify previously unknown variants and families. In this study, we will examine some of the most well-known research projects that use machine learning or deep learning to detect ransomware viruses.As a result, we ran studies to determine how malware evolution affects the study issues mentioned. We also talked about ransomware's new routes and how we expect it to evolve in the coming years, such as its expansion into IoT (Internet of Things), as IoT becomes more connected to infrastructures and people's homes.

Urooj et al. [7] conducted a study and investigation on ransomware, a notorious piece of software that has garnered reputation for its devastating and irreversible effects on its victims. The permanent damage caused by ransomware needs early identification of these attacks. The problem is approached in a novel method. This study provides information on datasets gathered from various sources and utilized in ransomware detection testing on multiple platforms.This study is also unusual in that it includes an overview of ransomware detection studies that use machine learning, deep learning, or a mix of the two techniques, as well as the advantages of dynamic analysis for ransomware detection. The study presented here examines ransomware detection investigations conducted between 2019 and 2021. This research provides a thorough list of future directions, laying the groundwork for future research. As a result, the goal of this research is to create a user manual that can be used by academics who want to work with existing technologies in the field of ransomware attack detection. It can help them create more effective ransomware detection algorithms while considering existing solutions.In the future, we will examine the role of static analysis in detecting ransomware attacks utilizing machine and deep learning technologies.

According to the problem presented in this research, Beaman et al. [8] caused a substantial spike in the number of ransomware attacks. Several institutions have been targeted, including healthcare, finance, and the government. There could be a multitude of reasons for the quick spike in attacks, but it appears that working remotely from home is one of them. Recent advances in ransomware analysis, detection, and prevention were studied as potential solutions to the problem. Honeypots, network traffic analysis, and machine learning-based methodologies were determined to be the primary targets of cutting-edge ransomware detection strategies.The studies also revealed that ransomware can be easily generated and used. Finally, we reviewed current research difficulties and future research objectives in the field of ransomware.

3.Theoretical Background

This section presents the history and efforts connected to the proposed solution in this study. Mobile cloud computing is a current and trending technology around the world, and it provides a number of benefits that improve the user experience. [2] Specialized features include storage, smartphone mobility via wifi or internet connection, and pay-as-you-go service. Similarly, Juniper Research reports on the growing use of mobile computing, stating that demand for cloud-based mobile apps in the public and commercial sectors is predicted to reach 9.5 billion dollars by 2014, with further increases expected in the near future.Similarly, the number of smartphone applications has increased in recent years, with apps in various areas such as entertainment, social media, internet streaming, banking, news, and so on. The primary reason for this is that mobile computing can offer the subscriber with a resource when and how they require it, based entirely on user organization.

According to a 2009 International Data Corporation (IDC) report, 74% of IT administrators and Chief Information Officers (CIOs) believe that user privacy issues are the most significant barriers to virtualization adoption. Mobile computing relies on three key components: technology, hardware, and communications. Clients can use gadgets such as smartphones and other portable devices.Consumers are rapidly adopting PDAs, thanks to the rapid expansion of wireless networks [3]. According to Allied Business Intelligence, more than 2.4 billion users would use mobile devices to access cloud computing platforms in 2015. Similarly, Google highlights several cloud-based products for consumers and organizations, including the Android OS, a critical item for mobile phones, as well as numerous software such as Google Maps, Streets, and others. Similarly, Google Stadia is a new technology that is a cloud-based gaming service that does not require any hardware but only requires an internet connection to connect [4]. The three fundamental methodologies for cloud computing are based on basic techniques used in the technology industry, such as parallelization, virtualization, and mass production.People who utilize mobile phones in cloud environments are subject to a wide range of external threats. Regular users and software developers should understand information privacy and authentication, since if they are aware of the privacy implications, they will not have any problems with hackers. People today are clueless of how to use technology and the advanced functionalities accessible on their cell phones and PDAs. Mobile security can be provided by a range of security aspects, including app installation, such as anti-virus software [1] [5].

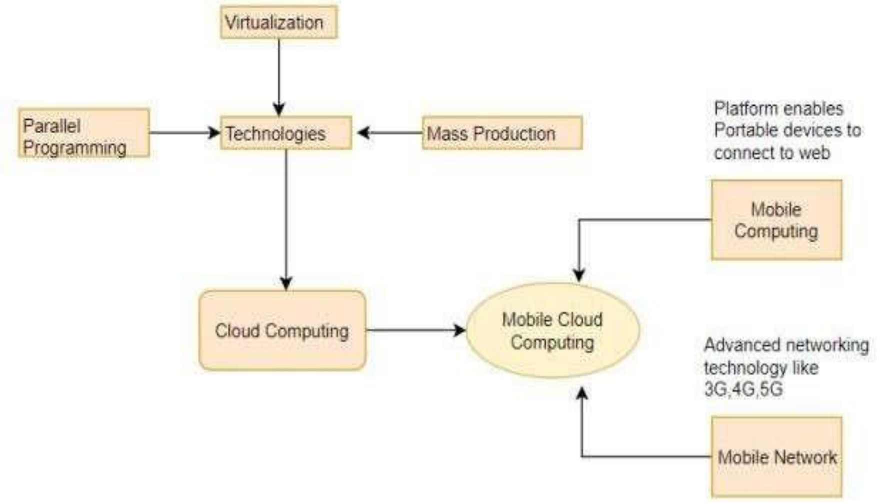
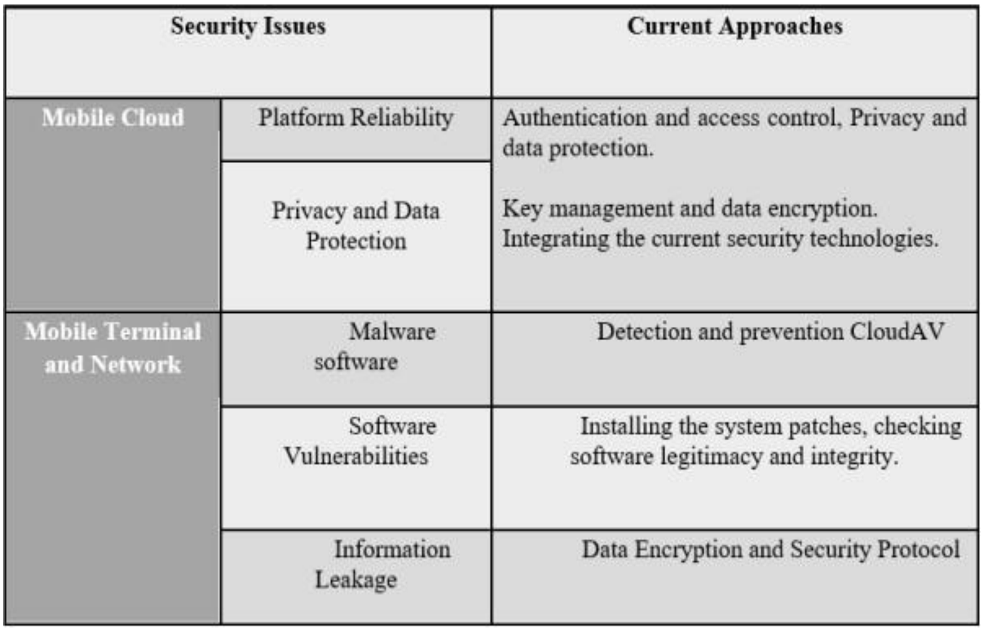


Figure 2: Architecture of Mobile Cloud Computing

Mobile Cloud Computing (MCC) security frameworks are classified into two types: application security and data security. Storing data in a virtual database without disclosing any information is more difficult for mobile users. An authentication method is used to ensure that when a user transmits a file to a cloud server for her ring with a large number of customers, the user seeing the file is a reliable client. Scalability is a network's ability to interact with clients seamlessly [11].Similarly, the most recent security technologies for online services, such as VPN usage, password encryption, authentication, and entry command, should be implemented to provide unit interrupted services against various threats, such as DOS attacks and data theft [9]. When such attacks occur, cloud services must provide a backup and restoration solution to restore client confidence. Figure 2 displays recent security concerns, while the table below details current options.



Security Issues		Current Approaches
Mobile Cloud	Platform Reliability	Authentication and access control, Privacy and data protection. Key management and data encryption. Integrating the current security technologies.
	Privacy and Data Protection	
Mobile Terminal and Network	Malware software	Detection and prevention CloudAV
	Software Vulnerabilities	Installing the system patches, checking software legitimacy and integrity.
	Information Leakage	Data Encryption and Security Protocol

Figure 2: List of Security Issues

An incursion is defined as any attempt to break the CIA of a device or network, and there are several sorts of intruder attacks. The most common are (DOS) assaults, which restrict legitimate consumers from using internet-based services. [6] An attacker in the virtual environment can send many attempts to authenticate VMs using cyborgs, preventing normal users from accessing them. Current Intrusion Detection and Prevention Systems (ID/PS) were unable to deliver the necessary level of protection and performance. Kumar employed a Fuzzy Mean Clustering-based artificial neural network to discover breaches in the cloud, where IDS is commonly implemented on end host cloud servers utilizing the following ways. [5] [7].

Authentication measures will prevent potential ransomware from using traditional signature-matching HIDS. Complex assessments based on existing IDS can be avoided by testing the controlled computer using the security procedure.[8] Signature matching procedures require proper monitoring, followed by a second layer of protection (Modiand Patel, 2013) that combines new NIDS technologies with older anomaly detection approaches to detect cyber-attacks in a network. Similarly, certain cloud protection services, such as Snort IDS, fail to distinguish between VM attacks that target everyone, from individual inhabitants to specific physical servers. Figure 3 depicts the various variants of cloud IDS[9]. A hybrid intrusion detection system improves the effectiveness of an identification device.Combining signature-based and anomaly-based approaches can greatly enhance this. The ability to endure fresh, unknown attacks by drawing on previously acquired information. Even though it has the same issues as all systems that use centralized control in a distributed situation [10], Wang et al developed a methodology that is similar to the central management approach.

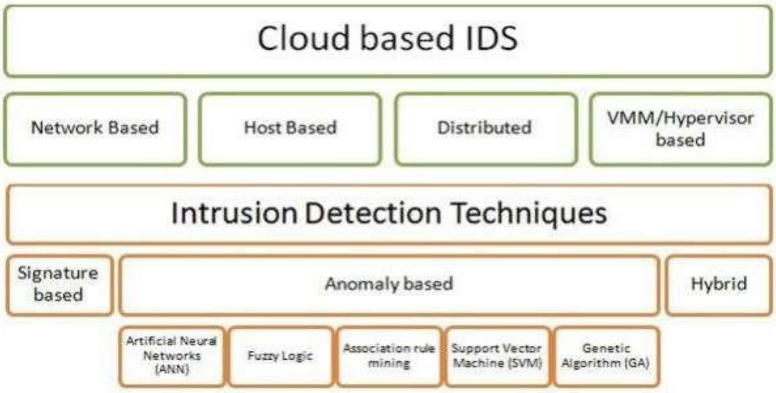


Figure 3: Structure of Intrusion Detection System in Cloud environment

Modi et al. also pioneered a progressive technique to intrusion detection. Hybrid IDS improves vulnerability security and performance by pre-processing packets and sending them to signature-based IDS after comparing them to previously recognized patterns. Previous systems were limited because they could not be effectively constructed to handle new types of assaults, which is also a time-consuming activity that takes far too long to investigate potential attacks.

Honeypots are a difficult concept in network security; such a device aims to collect information on infiltration attempts. There are various types of interaction honeypots accessible, ranging from minimal interaction honeypots that only imitate the communication layer to heavy interaction honeypots that run an entire operating system.One of the primary reasons for using a cloud service is to combine high and low communication honeypots in a cloud environment to evaluate assaults, and they must ensure that the scattered packets are valid after being relocated to HoneyCY during their transition to the cloud [12]. Similarly, it is composed of three design layers: HoneySrv collects honeypie devices and information, and HoneyV analyzes the virus obtained. Brown et al. described several virtualization systems used in honeypot sensors, whereas Saadi et al. provided an IDSf or smartphone device that incorporated Honeycomb, HoneyNet, and HoneyD honeypots [13] [14].

4. Proposed Model Analysis

A. Low Interaction Honey Pots

In an aggressive expansion, low-interaction honey pots are straightforward but efficient in detecting intruders, and the imitation of a low-interaction honey pot can be reduced with orders. One honey pot with little interaction is called Honeyd [5]. A minimal participation honeypot enables attackers to imitate services that require minimal user engagement. The goal of this kind of honeypot is to gather information from the initial phase of an attack due to the minimal level of interaction and system penetration. Information about the threat's motivation is rarely found.

The software for the virtual honeypot need an IP address in order to operate. To share a single run, multiple virtual honeypots will often utilize distinct IP addresses and network interfaces.As a result, the virtual honeypot is installed on a single physical server, and network address translation is done via a firewall or other means. The majority of high-interaction honeypots have the capacity to completely compromise the production system, whereas low-interaction honeypots are confined to simulating virtual systems.

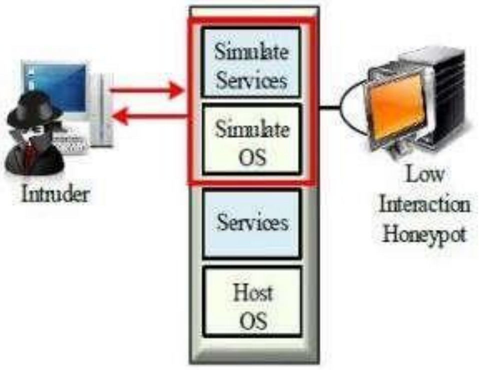


Figure 4: Low Interaction Honeypots

This is Honeyd's principal responsibility: most of their warnings are accurate and specific attack alarms. Honeys can detect activity on any port related to Transmission Control Protocol (TCP) or User Datagram Protocol (UDP), in addition to the fact that some actions are also written in ICMP (Internet Control Message Protocol). Additionally, they can fool the attacker by imitating specific components. The packets from the system response are appropriate for fingerprinting, which is done by using an application like Nmap to examine network traffic. Unluckily, a honeypot's attacker also makes use of services like Telnet, FTP, HTTP, POP3, and an SMTP (Simple Mail Transfer Protocol) server. In addition, they could have backdoors for viruses like Kuang2 and Mydoom inside of them.

B. High Interaction Honey Pots

In this study, we deploy and examine suspicious network data using a honeynet, while simultaneously developing multiple tools to support our analysis. We provide a web interface to monitor data collection, and we have a backend firewall to control outgoing connections from a possibly compromised honeypot in our architecture. Using a high-interaction honeypot host is one inexpensive strategy that mid-sized organizations commonly employ to benefit from a successful compromise that is both secure and hygienic. Some virtual machine alternatives for this environment are virtual PC, virtual box, XEN, VM ware, and user mode Linux.In order to obtain real network information and handle various types of analyzes, buffer overflows, and scans, we operate with a low interaction honeypot zone to reach the bases and genuine experiment results to have a high interaction honeypot. Additionally, we collaborate with a few actual computers to maintain our production server. Many recent research studies have focused on the use of honeypots to improve network security. According to Weiler's method, honeypots serve as a network shield that receives and routes all incoming traffic. Then follows instructions on how to sever that connection or establish a valid one.

This may not be the ideal course of action because the purpose of honeypots is to draw in attackers before they can be destroyed, not to serve as a protective or preventive measure. Teo introduces Japonica, a framework for solutions whose primary objective is to enable early and prompt response to unexpected threats through dynamic orchestration of attack detection, prevention, and reaction techniques. But until someone tries to access production systems both personally and professionally instead of depending on honeypots, there is always a significant danger of a false alert.

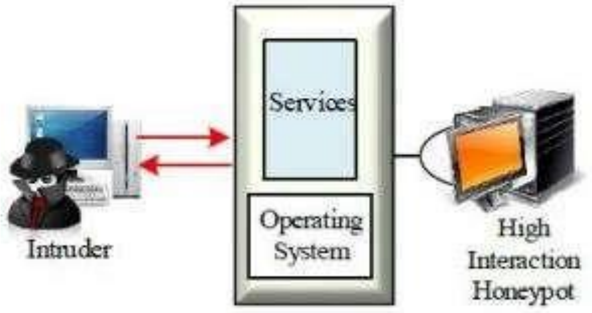


Figure 5: High Interaction Honeypots

In conclusion, a number of the previously suggested strategies made use of honeypots as a defensive technique to stop an attacker from attacking the network. This study provided a framework for interactive and enticing design with few usual defects, instead of blocking or defending systems, and a hybrid honeypot architecture that included low-interaction and high-interaction honeypots.

C. Hybrid Honey Pots

More sophisticated and scalable solutions have been developed by researchers and network security providers in response to the need for comprehensive information on a large number of IP addresses. Large-scale category architecture, also referred to as hybrid honeypot architecture, is the subject of these works. It is possible to employ and cultivate honeypots with both high and low interactions simultaneously, creating a coherent unit.

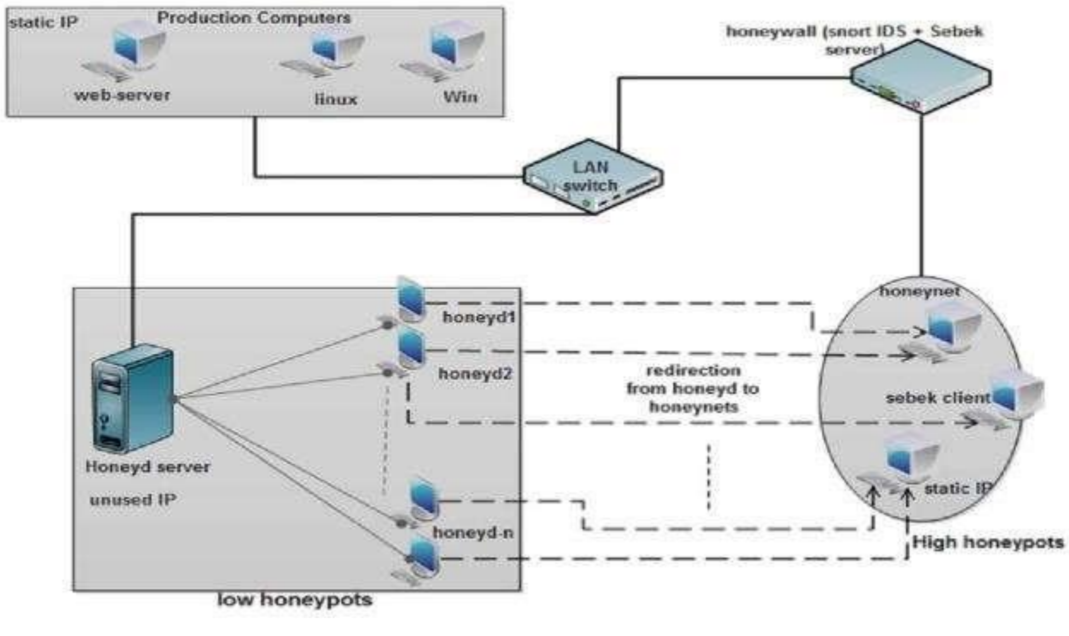


Figure 6: The proposed hybrid honeypot framework

A small number of high-interaction honeypots are needed to collect data, but combining honeypots with low and high interactions can potentially allow thousands of low-interaction honeypots to be deployed onto a host. Through the various layers of contact involved in this strategy, information from Internet dangers can be identified and obtained.

V. Honeypot Approached Model

A. Worms Activity

A software or program that, when installed on a honeypot, prompts other honeypots to modify their management to the point where they start forming links and generating connection or pair connection requests is referred to as a worm in the context of a network. The ability to distinguish between self-distributing network actions that restart the system and configure it in accordance with its own code and non-self-distributing network actions is made easier by this differentiation. It does not, however, plan to carry out the procedure on its own. Most worm variants have their own executable code, suggesting that the worms that were discovered had numerous connections and might have generated a password or overwritten a system buffer.Sincemost of these executables or viables are originally available as files and have nicknames that are generally strongly associated with them, the accompanying Table I. lists the different worm models together with the total number of worms that have been collected on our network. A hybrid honey pot that can record and collect both incoming and outgoing data while giving us control over our data is used in the proposed work to provide the best architecture that concentrates on the best decoy and lure architecture that can be absorbed by internal network attacks. Every action and operation carried out by the intruder is captured by the suggested honeynet and sent to a log for further examination.

B. Data Analysing Model

Examining the data collected from the original data is the data analysis module. Through internal honeypots, the honeynet gathers data, which is then sent for analysis. While we wait, we are sending the firewall logs to our analyzer and using a suitable firewall to obtain further information about the data we have already grabbed. Our back-end design will be able to access our production systems since, in the proposed architecture, a firewall module will function as a logger, recording all traffic and its status.

C. Honeynet Activity

Information control and information seizure, or data recording, are the two primary purposes of the honeynet, as previously mentioned. Preventing hackers from using the honeynet feature to access the other host is the primary objective of information control implementations. Obtaining complete control over an intruder is the aim of information seizure. Finding information in a covert manner without being discovered by spies is challenging. Secure Sockets Layer (SSL), IPsec (Internet Protocol Security), SSH (Secure Shell), and other similar routes are often encrypted by hackers. When conducting such actions, the data collection method must encrypt data using a specified account. Furthermore, to accomplish a multi-record level recording strategy, we use seizer tools on the honeypot with similar capacity [1].This technique can link the various steps that invaders take while also preventing the path from being defaulted to a single mechanism. Logs, data, and system activity captured by the honeypot's instruments are sent to the analyzing module. The information is kept as obtained data according to the characteristics and contents of the network connection. The recorded data from a honeynet is more dangerous and more dependable despite its modest size. By using the advantages of virtual technology—which is also utilized in honeynets—we may install a virtual honeypot [14] on a host. The expenses associated with developing honeynets are minimized and reduced with the help of this tactic. In spite of this, deploying a host requires more performance.

D. De-Militarized Zone

The De-Militarized Zone (DMZ) does not pass through various packets; it is not a network hardware device like a router or bridge [8]. The goal of the De-Militarized-Zone is to provide secure connection with servers before packets reach a firewall, eliminating the need for incoming firewall gaps between the deployed DMZ and the internal LAN or network. The security specifications for the DMZ's machines, networks, and peripherals are outlined in the policy. Workstations behind the firewall are able to remark on requests that are going to the DMZ in traditional De-Militarized-Zones. Machines in the DMZ reply to, attempt to forward, or resend inquiries from outside the public network or the internet. A server (such a proxy server) or several servers are frequently used as the computers that are deployed inside of demilitarized zones (DMZs).The firewall was installed after an attempt was made to prevent workstations in the DMZ from initiating inbound queries. Most computers on a regular LAN or an internal network operate behind the firewall in a DMZ configuration, enabling them to connect to the internet or other external networks. A few PCs or servers are positioned outside the firewall in the DMZ to establish the secure zone. These devices serve as an additional line of defense for the computers inside the firewall zone by intercepting traffic and agent queries for other network segments. Typically, a DMZ is home to servers that offer clients a range of internet-based services. Among the services offered are DNS server, FTP, SMTP, IMAP4, and POP3.Even though these servers must be connected to limited internet access, they can also safeguard the firewall. The servers and honeypots could be in the DMZ or inside the network, although the DMZ is recommended. The best structure we are looking for that has been shown in Fig. 7.

E. Proposed Hybrid Honeypot Framework

In order to alter organizational, financial, and crucial conducted server zone networks, the suggested innovation presents a versatile honeypot-based network security system that makes use of the active, dynamic deployment and configuration of hybrid honeypots. Exploiting free, ready-to-use IP addresses offered by operating systems or distributed systems and their services is the basic concept of low-interaction honeypots. On production servers connected to a network, they replicate dispersed operating systems and related services. Honeypots that receive network traffic are typically designed to steer traffic to high interaction honeypots, which present targeted services to attackers. The use of hybrid or half-breed technology to tackle honeypot technology falls into two primary categories:Due to the quantity of honeys and their unique service settings that are automatically determined by the network authority, using minimal administrative problems The demand for high interaction honeypots, or honeynets, in the network as demonstrated by the scenario where traffic is redirected from a low interaction honeypot shows how attractive honeys are to attackers as real systems.

F. Proposed Honeynet

As a result of phoney machines in the network, the system administrator must first designate the IP addresses of the actual honeypot or important host within the honeynet, allow traffic redirection from low interaction honeypots, and report any attacker activity. Not only is the communication path between devices altered by the locution redirection. More emphasis was placed on reformatting incoming network packets to a certain honeyed format and reintroducing them to the network. If they make their deployment in this way, they will be able to find the real honeypot. The invader is then given the impression that he is communicating with a genuine computer when he replies to him.We aim to provide an example of typical Local Area Network behavior for our hybrid honeypot technique. Figure 7 shows the deployment and position. The primary switch and other production systems are directly connected to a honeypot server depicted in this diagram, which has limited interaction. Additionally, it shows the real honeypot in the architecture honeynet, which is prepared to take in network traffic that comes directly from the minimal interaction honeypot or traffic that is diverted from it. The architecture shows that although the low contact honeypot devices seem to be a physical or production system, they are actually virtual machines from the beginning. Network Address Translation (NAT) is something we might use in our architecture.By employing this technique, it is not necessary to change the configuration of each honeypot to be dynamically in an internal home for external homes that are retrieved through Network Traffic Monitoring (NTM). Consequently, we should note that we can also omit this step if we configure the honeypot to support dynamic address reconfiguration.

The low-interaction honeypot server depicted in this diagram has three major capabilities that are implemented in separate threads. The initial honeys server uses a network scanning application to gather information on the various operating systems in the network, their specific direct or administrate ports, and running services, and then collects and saves this information in a file. The next thread reads the data from the file and updates the low interaction honeypot setup as needed. As a result, in the real network component, it includes the operating system, their services, and port and network support distribution. The last thread examines the honeypot log traffic data with little involvement and saves it to a certain file. Furthermore, while attackers are engaged, the servers wait for arriving traffic to flow to unused IP addresses and then supposed to identify those IPs. A programming language, network scanning tools, and operating system must be chosen to build the proposed system depicted in Fig. 7. Even though the approach architecture framework is progressed in general and not limited to a certain preference. Due to the suppleness of its availability to deploy the security application, the operating system chosen for the honeys server needed to perform open source. For this, we used the Linux Fedora 12.0 edition, which offers the required functionality thanks to our framework. The programming language required network library availability language functionality as well as the ability to easily integrate Fedora resources. In such circumstances, we use the Python programming language, which includes the required networking package.

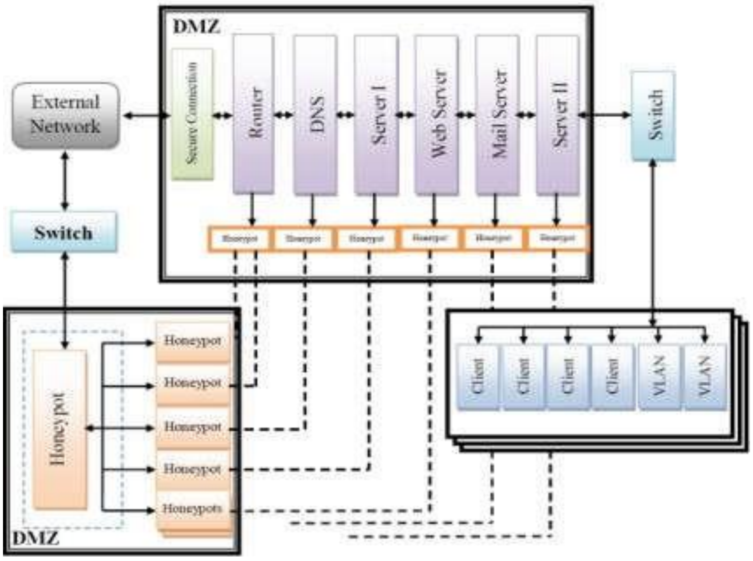


Figure 7: Hybrid honeypot architecture

A network scanning tool was then used to determine the type of operating system used in the network as well as the various ports in the production network, and to offer the necessary information. Nmap was picked as a component of our project for a specific purpose. This network tool can be used in two different active modes to acquire information about the many distributed operating systems that are available, as well as conduct ports and assumed running services in the network. These two folded are regular mode, which means that this tool gathered data at a particular time. In this mode, Nmap tries to parallelize port scans, even though information can be collected in the shortest time possible. However, the server may become overburdened with input or output data, and network traffic may increase as a result. The polite mode of the Nmap tool is the second mode, which gathers information slowly. The utility serializes port scans with delaying between sequential scans in this mode. This scenario applies to a machine and a network that are friendly in terms of time consumption and requiring a long time to complete scans. Regardless, we shall do a thorough network scan.

As shown in Fig. 8, the Nmap scan procedure involves issuing a ping to establish all devices on the network and collect their IP addresses, although not in a permanent file. This file can be utilized for the next scan, which will include operating system or port scans for the IP addresses that have been discovered. The results of the scans are logged into a specific file, which is often an XML file that is evaluated every time until the scan is completed. When the tool scans are completed and halted, an analyst starts and runs in a thread to extract the acquired data from the file, which creates an automatic profile to store the data.

G. Deployment of Honeypots

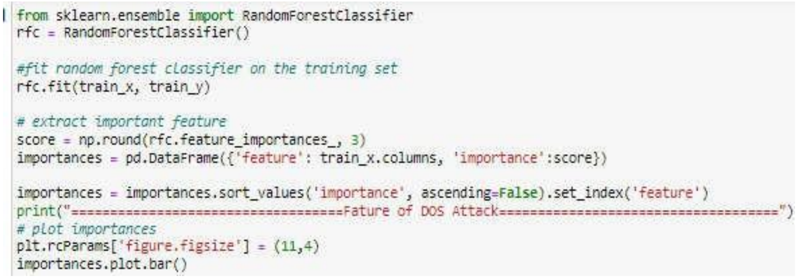
The basic idea behind a hybrid honeypot is to use unused IP addresses, but there is a problem that helps to answer how to isolate them from the current operating system and therefore reduce the risk of revealing the real and production hosts in the network and allowing them to be hacked by intruders. A simple forward approach was implemented to ensure a steady continuance after integrating virtual systems into the production system via operating system distribution, and this should be done while removing physical honeypots. A honeypot is an internet platform that acts as a decoy to lure cyber attackers and detect, divert, and investigate efforts to gain unauthorised access to networks. Operating a honeypot can be costly, in part because of the specialized expertise required to design and maintain a system that aims to expose an organization's system resources while preventing attackers from gaining access to any operational systems. Honeynet is an example of a honeypot with a lot of interaction. It can be described as a network containing multiple systems. Honeynet can link together comprehensive information on hackers or attackers, such as keystrokes used to break into a system, chat sessions with other attackers, and the tools they employ to probe and exploit vulnerable systems. This information can reveal a great deal about the attacker. The hybrid honeypot system combines low-level and high-level interactive honeypot systems. We employed a variety of machine learning approaches, including Random Forest, Support Vector Machine, and K-Neighbours, and finally Ensemble Learning, which is a blend of various machine learning techniques.

The categorization problems are solved using logistic regression. Because it can generate probabilities and classify new data using both continuous and discrete datasets, logistic regression is a key machine learning approach. Logistic regression can be used to categories observations based on many forms of data and can quickly identify the most useful factors for classification. Every Decision Tree has some elements, such as nodes, which are the sites where it splits, and edges, which are the outcomes or results when it splits. The nodes that perform the first splits are called roots, whereas the elements that forecast the eventual outcome are called leaves. One of the most common machine learning issues is support vector machines (SVMs). SVMs are versatile in that they may be utilized for both classification and regression tasks. Linear SVM is a classifier for linearly separable data, which means a dataset can be classified into two classes using a single straight line, and the classifier is called Linear SVM. Non-linear SVM is a classifier that is used for non-linearly separated data. This means that if a dataset cannot be classified using a straight line, it is non-linear data, and the classifier used is termed Non-linear SVM. In n-dimensional space, there can be several lines/decision boundaries to separate the classes, but we must choose the optimum decision boundary to help classify the data points. The hyper plane of SVM refers to the best boundary. The hyper plane's dimensions are determined by the features in the dataset; for example, if there are two features (as shown in the image), the hyper plane will be a straight line. Hyper plane will be a two-dimensional plane if there are three features.

The following are the general procedures for creating a K nearest neighbor’s algorithm: Calculate the Euclidean distance between the new data point x and all the existing data points in the data collection. Sort the points in the data set in ascending order of distance from x. Predict using the same classification as the majority of the K data points closest to x. An extremely low K value, in particular, will cause your model to flawlessly predict your training data while poorly predicting your test data. Similarly, a K number that is too high will make your model needlessly complicated. We implement the Random Forest Classifier using the Python computer language's Scikit Learn package and the IRIS dataset, which is a well- known dataset.The Random Forest, also known as the Random Decision Forest, is a supervised Machine Learning technique that uses decision trees to do classification, regression, and other tasks.A random subset of the training set is used by the Random Forest classifier to generate a collection of decision trees. It consists of a set of decision trees (DT) derived from a randomly selected subset of the training set, which then accumulates votes from various decision trees to get the final prediction. A group of elements considered rather than individually is referred to as an ensemble. Ensemble solves a problem by creating several models and combining them.Ensemble approached said in the model's resilience and generalizability.In this post, we'll look at certain techniques and their Python implementations. Ensemble learning is the act of intentionally generating and combining many models, such as classifiers or experts, to solve a computer intelligence problem. Ensemble learning is mostly used to boost performance (classification, prediction, function approximation, etc.)

6. DOS Attack Detection Using Honeypot Machine Learning

A DoS attack is one that attempts to bring a machine or network to a halt, rendering it unreachable to its intended users. DoS attacks work by flooding the target with traffic or transmitting information that causes it to crash. The DoS attack deprives genuine users (workers, members, or account holders) of the service or resource they expected in both cases. DoS attacks frequently target high profile institutions including banks, commerce, and media companies, as well as government and trade organisations. DoS attacks rarely result in the theft or loss of sensitive data or other assets, but they can cost the victim a lot of time and money to cope with. DoS attacks can be carried out in two ways: flooding or crashing systems. Flood attacks happen when a system receives too much traffic for the server to buffer, slowing it down and eventually stopping it. A random forest is a meta estimator that employs averaging to increase predicted accuracy and control over fitting by fitting a number of decision tree classifiers on various sub-samples of the dataset. If bootstrap True (default), the sub-sample size is regulated by the max sample's argument, otherwise, the entire dataset is utilised to create each tree.



```
from sklearn.ensemble import RandomForestClassifier
rfc = RandomForestClassifier()

#fit random forest classifier on the training set
rfc.fit(train_x, train_y)

# extract important feature
score = np.round(rfc.feature_importances_, 3)
importances = pd.DataFrame({'feature': train_x.columns, 'importance':score})

importances = importances.sort_values('importance', ascending=False).set_index('feature')
print("==================================Fature of DOS Attack====================================")
# plot importances
plt.rcParams['figure.figsize'] = (11,4)
importances.plot.bar()
```

Figure 8: Future of DOS Attack

```
===================================Fature of DOS Attack================================
<AxesSubplot:xlabel='feature'>
```

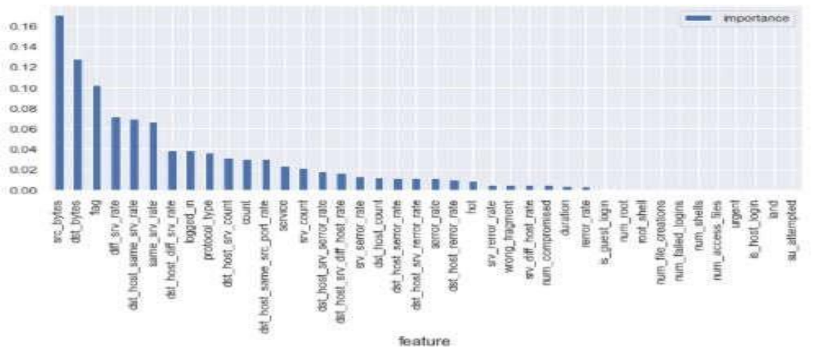
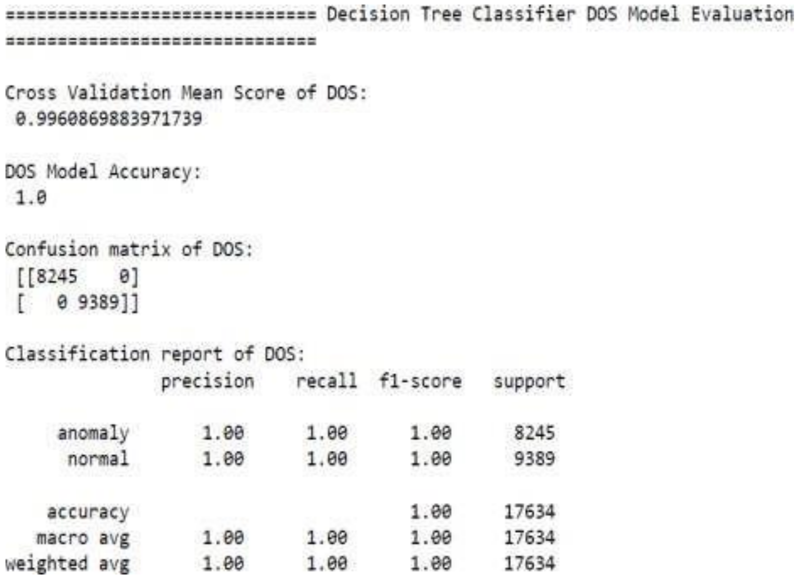


Figure 9: DOS Attack Detection



```
============================== Decision Tree Classifier DOS Model Evaluation
==============================

Cross Validation Mean Score of DOS:
 0.9960869883971739

DOS Model Accuracy:
 1.0

Confusion matrix of DOS:
 [[8245    0]
 [   0 9389]]

Classification report of DOS:
              precision    recall  f1-score   support

     anomaly       1.00      1.00      1.00      8245
      normal       1.00      1.00      1.00      9389

    accuracy                           1.00     17634
   macro avg       1.00      1.00      1.00     17634
weighted avg       1.00      1.00      1.00     17634
```

Figure 10: Decision Tree Classifier DOS Model Evaluation

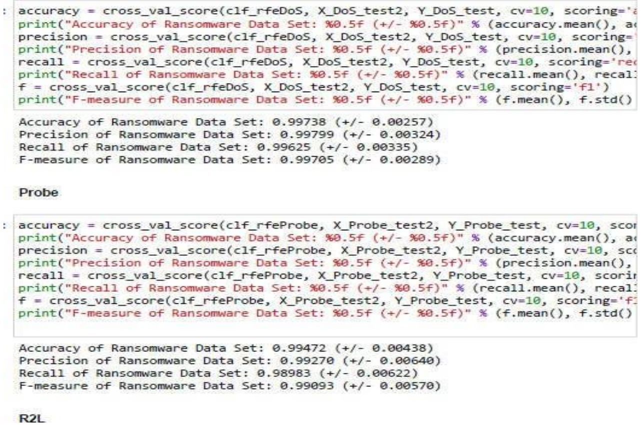
We implement the Random Forest Classifier using the Python computer language's Scikit Learn package and the IRIS dataset, which is a well- known dataset.Random forest, also known as Random Decision Forest, is a supervised Machine Learning technique that uses decision trees to do classification, regression and other tasks.

A random subset of the training set is used by the Random Forest classifier to generate a collection of decision trees.It consists of a set of decision trees (DT) drawn from a randomly selected subset of the training set, which then gathers votes from various decision trees to determine the final prediction.

7. Ransomware Attack Detection using Honeypot Machine Learning

Ransomware is a type of malware that encrypts the files of its victims. The attacker then demands a ransom from the victim in exchange for restoring access to the data. There are several ways ransomware might get access to a computer. Phishing spam-attachments that arrive in an email disguised as a file the victim should trust are one of the most popular delivery tactics. They can take over the victim's computer once they have been downloaded and opened, especially if they contain built in social engineering techniques that deceive people into granting administrative access.Other, more aggressive ransomware, such asNotPetya, takes advantage of security flaws to infect machines without the need to deceive people.There are various methods through which ransomware criminals select the firms they attack. It is sometimes a matter of timing: for example, attackers may target universities since they have smaller security teams and a diverse user base that shares a lot of files, making it easier to breach their defences.

Honeypot is a network attached system that hackers employ as a trap to detect and investigate the methods and sorts of attacks they utilize.It operates as a prospective internet target and alerts the defenders to any unauthorized attempts to access the information system. Large firms and organizations concerned in cyber security utilize honeypots the most. It assists cyber security researchers in learning about the many types of attacks employed by criminals and attackers. It is suspected that even the cybercriminals use these honeypots to decoy researchers and spread wrong information. Classification Accuracy is what we usually mean, when we use the term accuracy. It is the ratio of number of correct predictions to the total number of input samples. The precision is the ratio tp / (tp + fp) where tp is the number of true positives and fp the number of false positives. The precision is intuitively the ability of the classifier not to label as positive a sample.



```
accuracy = cross_val_score(clf_rfeDoS, X_DoS_test2, Y_DoS_test, cv=10, scoring='a
print("Accuracy of Ransomware Data Set: %0.5f (+/- %0.5f)" % (accuracy.mean(), ac
precision = cross_val_score(clf_rfeDoS, X_DoS_test2, Y_DoS_test, cv=10, scoring='
print("Precision of Ransomware Data Set: %0.5f (+/- %0.5f)" % (precision.mean(),
recall = cross_val_score(clf_rfeDoS, X_DoS_test2, Y_DoS_test, cv=10, scoring='rec
print("Recall of Ransomware Data Set: %0.5f (+/- %0.5f)" % (recall.mean(), recal
f = cross_val_score(clf_rfeDoS, X_DoS_test2, Y_DoS_test, cv=10, scoring='f1')
print("F-measure of Ransomware Data Set: %0.5f (+/- %0.5f)" % (f.mean(), f.std()

Accuracy of Ransomware Data Set: 0.99738 (+/- 0.00257)
Precision of Ransomware Data Set: 0.99799 (+/- 0.00324)
Recall of Ransomware Data Set: 0.99625 (+/- 0.00335)
F-measure of Ransomware Data Set: 0.99705 (+/- 0.00289)
```

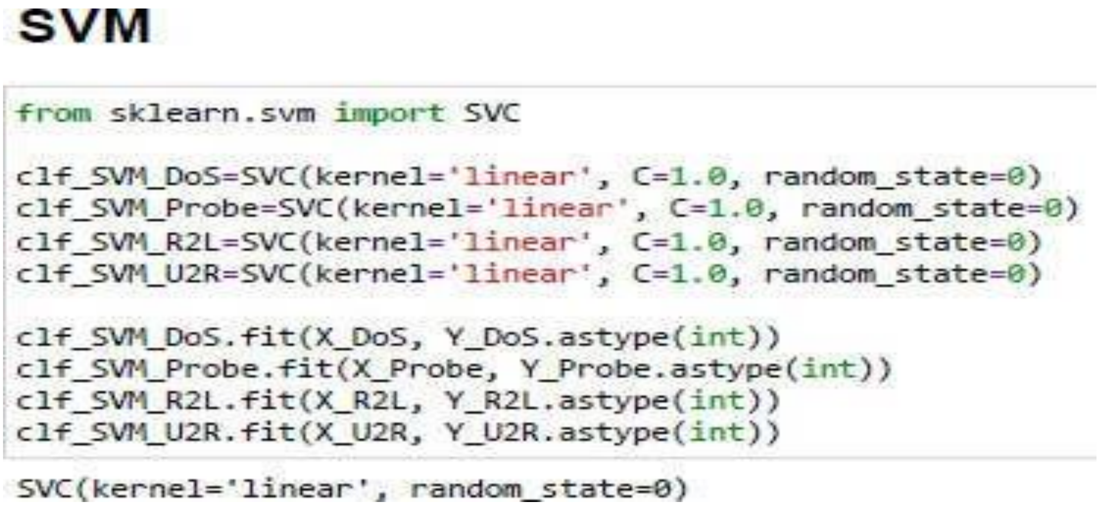
Probe

```
accuracy = cross_val_score(clf_rfeProbe, X_Probe_test2, Y_Probe_test, cv=10, scor
print("Accuracy of Ransomware Data Set: %0.5f (+/- %0.5f)" % (accuracy.mean(), ac
precision = cross_val_score(clf_rfeProbe, X_Probe_test2, Y_Probe_test, cv=10, sco
print("Precision of Ransomware Data Set: %0.5f (+/- %0.5f)" % (precision.mean(),
recall = cross_val_score(clf_rfeProbe, X_Probe_test2, Y_Probe_test, cv=10, scorin
print("Recall of Ransomware Data Set: %0.5f (+/- %0.5f)" % (recall.mean(), recal
f = cross_val_score(clf_rfeProbe, X_Probe_test2, Y_Probe_test, cv=10, scoring='f1
print("F-measure of Ransomware Data Set: %0.5f (+/- %0.5f)" % (f.mean(), f.std()

Accuracy of Ransomware Data Set: 0.99472 (+/- 0.00438)
Precision of Ransomware Data Set: 0.99270 (+/- 0.00640)
Recall of Ransomware Data Set: 0.98983 (+/- 0.00622)
F-measure of Ransomware Data Set: 0.99093 (+/- 0.00570)
```

R2L

Figure 11: Accuracy, Precision, Recall Results for Machine Learning Algorithm.



SVM

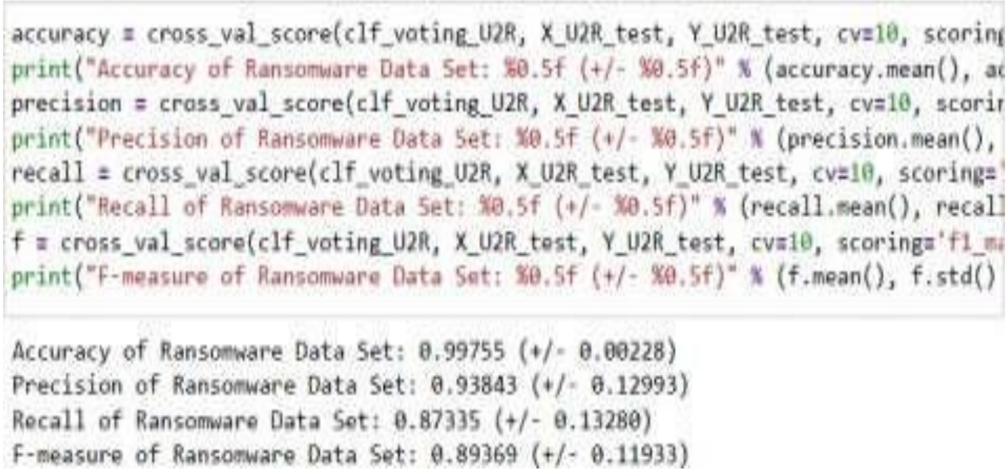
```
from sklearn.svm import SVC

clf_SVM_DoS=SVC(kernel='linear', C=1.0, random_state=0)
clf_SVM_Probe=SVC(kernel='linear', C=1.0, random_state=0)
clf_SVM_R2L=SVC(kernel='linear', C=1.0, random_state=0)
clf_SVM_U2R=SVC(kernel='linear', C=1.0, random_state=0)

clf_SVM_DoS.fit(X_DoS, Y_DoS.astype(int))
clf_SVM_Probe.fit(X_Probe, Y_Probe.astype(int))
clf_SVM_R2L.fit(X_R2L, Y_R2L.astype(int))
clf_SVM_U2R.fit(X_U2R, Y_U2R.astype(int))

SVC(kernel='linear', random_state=0)
```

Figure 12: Using SVM Classifier in Honeypot.



```
accuracy = cross_val_score(clf_voting_U2R, X_U2R_test, Y_U2R_test, cv=10, scoring
print("Accuracy of Ransomware Data Set: %0.5f (+/- %0.5f)" % (accuracy.mean(), ac
precision = cross_val_score(clf_voting_U2R, X_U2R_test, Y_U2R_test, cv=10, scorir
print("Precision of Ransomware Data Set: %0.5f (+/- %0.5f)" % (precision.mean(),
recall = cross_val_score(clf_voting_U2R, X_U2R_test, Y_U2R_test, cv=10, scoring='
print("Recall of Ransomware Data Set: %0.5f (+/- %0.5f)" % (recall.mean(), recall
f = cross_val_score(clf_voting_U2R, X_U2R_test, Y_U2R_test, cv=10, scoring='f1_ma
print("F-measure of Ransomware Data Set: %0.5f (+/- %0.5f)" % (f.mean(), f.std()

Accuracy of Ransomware Data Set: 0.99755 (+/- 0.00228)
Precision of Ransomware Data Set: 0.93843 (+/- 0.12993)
Recall of Ransomware Data Set: 0.87335 (+/- 0.13280)
F-measure of Ransomware Data Set: 0.89369 (+/- 0.11933)
```

Figure 13: accuracy, precision, Recall, F-measure

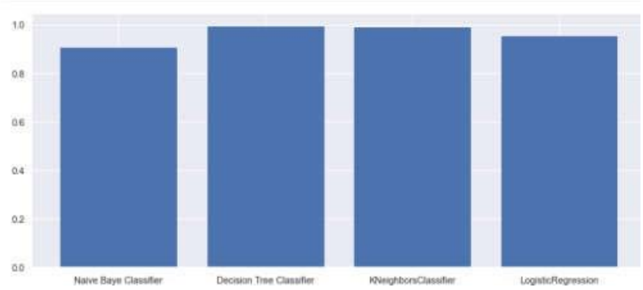


Figure 14: Comparison using Different Machine learning Algorithms

8. CONCLUSION

The suggested hybrid honeypot architecture system protects production systems to some extent. It accomplishes this by lowering the chance of hacker activity and targeting our production systems through the use of lure systems in the network, which prevent the hacker from learning about these systems, their state, or his fingerprint, and so mistaking the p honey systems for real systems. Without the redirection capabilities, we will not be able to achieve our goal, and the production system would remain exposed to direct attacks that do not travel via the controlled honeypot system. In the suggested approach, production honeypots can only perform a passive role, logging different attacker activities so that the system administrator can extract and analyze them using data mining. This could play a more active role by evaluating the attacker's activity and reducing the different types of attacks by using a signature file or a signature database that can build and my data. As we have seen, honeypots will be able to add and release warnings, as well as send notifications to the administrator, the intruder kind, and several possible solutions to stop the attack from spreading. There are some observations have been identified for the future prospective:

- Hybrid Honeypot System can be used for in-depth analysis of various attacks and for capturing various malware attacks.
- Hybrid Honeypot System can be implemented in a cloud computing to reduce the security risk.
- Hybrid honeypot system is moderate complexity and it can be used to obtain more precise information of the intruder.
- Hybrid Honeypot System can modernize the present system and style of the new solutions for identification of user behavior.
- Hybrid Honeypot can be made self- learning i.e. it can monitor local network and can dynamically deploys virtual honeypots based on network makeup.

REFERENCES

[1] Sagar Pande, Aditya Khamparia, Deepak Gupta, and Dang N. H. Thanh,” DDOS Detection Using Machine Learning Technique”, October 2020.

[2] Arshi M, Nasreen MD, and Karanam Madhavi,” A Survey of DDOS Attacks Using Machine Learning Techniques”, E3S Web of Conferences 184, 01052 (2020).

[3] Jiangtao Pei, Yunli Chen, Wei Ji, “A DDoS Attack Detection Method Based on Machine Learning”, IOP Conf. Series: Journal of Physics: Conf. Series 1237 (2019) 032040.

[4] Swathi Sambangi * and Lakshmeeswari Gondi,” A Machine Learning Approach for DDoS (Distributed Denial of Service) Attack Detection Using Multiple Linear Regression”, 25 December 2020.

[5] Tuğba Aytaç, Muhammed Ali Aydın, Abdül Halim Zaim,” Detection DDOS Attacks Using Machine Learning Methods”, Electrica, 2020; 20(2): 159-167.

[6] DamienWarren Fernando, Nikos Komninos and Thomas Chen,” A Study on the Evolution of Ransomware Detection Using Machine Learning and Deep Learning Techniques”, IoT 2020, 1, 551–604; doi:10.3390/iot1020030

[7] Urooj, U.; Al-rimy, B.A.S.; Zainal, A.; Ghaleb, F.A.; Rassam, M.A. Ransomware Detection Using the Dynamic Analysis and Machine Learning: A Survey and Research Directions. Appl. Sci. 2022, 12, 172. https://doi.org/10.3390/ app12010172.

[8] Craig Beaman, Ashley Barkworth, Toluwalope David Akande, Saqib Hakak, Muhammad Khurram Khan,” Ransomware: Recent advances, analysis, challenges and future research directions”, 24 September 2021.

[9] Khattab M, Sangpachatanaruk C, Mosse D, MelhemR, Znati T. Roaming honeypots formitigating service-level denial-of-service attacks. In: Proceedings of the IEEE 24th international conference on distributed computing systemsMarch, p. 328–37, 2004.

[10] Krawetz N. Anti-honeypot technology. IEEE Security & Privacy Magazine, Vol. 2(1), pp. 76–9, 2004.

[11] Kuwatly I, Sraj M, Al-Masri Z, Artail H. A dynamic honeypot design for intrusion detection. In: Proceedings of IEEE/ACS international conference on pervasive services, p. 95–104, July 2004. [14] Lok Kwong Yan. “Virtual honeynetsrevisited,” Information Assurance Workshop, pp 232-239, 2005.

[12] Omid Mahdi Ebadati E., Kaur H., Alam A.M., “A Secure Confidence Routing Mechanism Using Network-based Intrusion Detection Systems”, OLS Journal of Wireless Information Networks & Business Information System, Open Learning Society, Nepal, pp. 83 – 93, 2010.

[13] Teo L, Sun A, AhnJ. Defeating internet attacks using risk awareness and active honeypots. In: Proceedings of the second IEEE international information assurance workshop, p.p. 155–67, April 2004. Virtual PC, http://www.microsoft.com/windows/products/win family/virtualpc/ default.mspx, 2008.

[14] Weiler N. Honeypots for distributed denial of service attacks. In: Proceedings of the 11th IEEE international workshop on enabling technologies: infrastructure for collaborative enterprises (WETICE’02) June 2002.

[15] Yeldi S., Gupta S., Ganacharya T., Doshi S., Bahirat D., Ingle R., et-al. Enhancing network intrusion detection system with honeypot. Conference on Convergent Technologies for Asia- Pacific Region TENCON 2003, p. 1521–6, October 2003.

[16] Faizan Ullah, Qaisar Javaid, Abdu Salam, MasoodAhmad, NadeemSarwar, Dilawar Shah, and Muhammad Abrar,” Modified Decision Tree Technique for Ransomware Detection at Runtime through API Calls”, 1 August 2020.

[17] Samuel Egunjobi, Simon Parkinson, and Andrew Crampton,” Classifying Ransomware Using Machine Learning Algorithms”, Department of Computer Science, School of Computing and Engineering, University of Hudders eld, Queensgate, Hudderseld HD1 3DH, UK.