R2NET: Storage and Analysis of Attack Behavior Patterns

M. R. Amal[1*], and P. Venkadesh[2]
[1] Department of Computer Science and Engineering,
Noorul Islam Centre for Higher Education, Tamil Nadu 629180, India
[e-mail: amalmr589@gmail.com]
[2] Department of Computer Science and Engineering,
Noorul Islam Centre for Higher Education, Tamil Nadu 629180, India
[*]Corresponding author: M. R. Amal

Received August 3, 2022; revised September 8, 2022; accepted November 9, 2022; published February 28, 2023

Abstract

Cloud computing has evolved significantly, intending to provide users with fast, dependable, and low-cost services. With its development, malicious users have become increasingly capable of attacking both its internal and external security. To ensure the security of cloud services, encryption, authorization, firewalls, and intrusion detection systems have been employed. However, these single monitoring agents, are complex, time-consuming, and they do not detect ransomware and zero-day vulnerabilities on their own. An innovative Record and Replay-based hybrid Honeynet (R2NET) system has been developed to address this issue. Combining honeynet with Record and Replay (RR) technology, the system allows fine-grained analysis by delaying time-consuming analysis to the replay step. In addition, a machine learning algorithm is utilized to cluster the logs of attackers and store them in a database. So, the accessing time for analyzing the attack may be reduced which in turn increases the efficiency of the proposed framework. The R2NET framework is compared with existing methods such as EEHH net, HoneyDoc, Honeynet system, and AHDS. The proposed system achieves 7.60%, 9.78%%, 18.47%, and 31.52% more accuracy than EEHH net, HoneyDoc, Honeynet system, and AHDS methods.

Keywords: Cloud computing, Honeynet, Machine learning, R2NET, Zero-day attacks.

1. Introduction

Cloud computing has become a significant source of data storage for all businesses. Data storage demands are growing every day, making cloud computing inevitable [1]. With an increasing number of people using the services, there is an increased risk of attacks and security vulnerabilities that could compromise or cause the loss of data [2]. The primary motivations for intrusion operations that target data systems are fame, reputation, financial gain, and national community benefit [3].

Many technologies or software are employed in computer network systems to secure corporate or personal data [4]. The tools employed for this objective include intrusion detection and prevention systems, vulnerability scanners, antivirus programs, and firewalls [5,6]. However, as a single monitoring agent, these tools are complex and time-consuming, and thus fail to identify ransomware [7,8]. To detect ransomware, a honeypot system is used.

A honeypot is a cybersecurity prevention tool that is compromised and located in a critical network area to gather data about potential attackers [9]. Honeypots are a new technology with significant potential for network security. They can be used within, outside, and even inside of the firewall [10,11]. This is a system of traps designed to direct attackers and hackers away from critical resources so they cannot access them. It can also be employed to study the frameworks and tools employed by an attacker [12].

Honeypots can be used to provide a security feature by diverting the attacker away from the actual transaction, enabling us to obtain the attacker's details [13,14]. A second benefit is that honeypot data can be used to improve our security system [15]. Therefore, a record and replay-R2NETbased hybrid honeynet (R2NET) system is proposed in this paper.

The R2NET system is based on the virtual machine's record and replay (RR) foundation. By combining the hybrid honeynet system with RR, the system allows for fine-grained analysis by deferring time-consuming analysis until the replay stage. Here, the logs of the attackers are clustered using machine learning techniques and thus increase the efficiency of the R2NET system. These approaches combine to create an efficient and versatile honeypot system.

The remaining section of the proposed framework is organized as follows. Section II describes the literature survey in detail. Section III describes the proposed R2NET system. Section IV describes the results and discussion. Section V describes the conclusion and future work.

2. Literature Review

A honeypot is not used to address network security issues alone; rather, it is utilized in the context of a system's security, so honeypots are built and placed to address the challenges they are supposed to solve. In this part, we examined some new initiatives addressing cloud security concerns and mitigating attacks in the cloud computing environment.

In 2018, N.K. Bao *et al* [16] have proposed a novel honeynet architecture called the Efficient Elastic Hybrid Honeynet. Using this system, attack traffic response times can be sped up, compromised honeypots can be efficiently isolated, honeypot fingerprinting can be eliminated, and resources for maintenance and deployment can be optimized. In testing the system with real attack traffic, they found that it is not only practical but also extremely efficient.

In 2019, Fans, W., *et al*. [17] proposed HoneyDOC, a new honeypot architecture that uses the Decoy-Orchestrator-Captor perspective to dissect and decouple the honeypot, as

represented by the strong SDN-enabled framework, to facilitate all-round honeypot design and deployment. Real data is used to demonstrate the effectiveness of data reduction and traffic redirection for data analysis. The Experimental analysis suggest that the proposed architecture is both feasible and efficient.

In 2019, Marydas, M., and Varsha Priyah, J.N., [18] proposed a Honeynet system in their cloud network. This system detects attacks or suspicious activity on protocols such as SSH, FTP, and others. Heuristically, these strategies are useful for distinguishing between normal and malicious traffic. To analyze Honeynet's documented activities, they used three machine learning techniques. They used Nave Bayes, SVM, and Random Forest to train the models so that new data from the honeynets can be classified with greater accuracy as malicious.

In 2019, Saxena, M.A., *et al* [19] proposed a cloud-based solution for a high-engagement honeypot with EFS (Elastic File System), VPN (Virtual Private Network), VPC (Virtual Private Cloud), and Kerberos as a service to provide network/data security. It would be easy to operate, safe, and cost-effective. The method improves the confidentiality, availability, security, and integrity of the system.

In 2019, S.K. Sood and K.D. Singh [20] proposed a safe framework that not only detects harmful OFDs but also decreases false alerts. A virtual honeypot buried in the optical fog layer (POS) is shifted using an intrusion detection system and a hidden Markov model to detect the malicious one. According to the results, the suggested system significantly decreases false alarms, recognises harmful ones, and switches them to the virtual honeypot efficiently.

In 2020, Al-Mohannadi, H., *et al*. [21] developed a threat intelligence approach for examining attack information through cloud-based web services to support active threat intelligence. Cyberattacks on the honeypot system provide critical intelligence that can be employed in systems like IDS, IPS, and firewalls to secure an organization's production. On the other hand, attack intelligence does not only assist in detecting threats but also in identifying how they are carried out by analysing the actions.

In 2020, E.M. Kandoussi, *et al* [22] proposed a hybrid security system that combines virtual machine migration with honeypots. It examines security policies concerning their effectiveness. Additionally, our proposed approach quantifies potential attack paths before categorizing them into 2 groups: attack paths just examined and attack paths studied and exploited based on black box intrusion stages. Based on the attack graph and stochastic game theory, the attacker-defender interaction is modeled.

In 2020, Kong, T., *et al*., [23] offered AHDS, an Automated Honeynet Deployment Strategy, which aims to automate honeynet deployment strategy enhancement by examining system structure variation. A network attack graph has been proposed to encompass and model network attacks in depth. AHDS results in 83 percent lower attack success rates in container-based clouds while also being flexible and adaptable for large-scale implementations.

In 2021, Singh, K.D. [24] developed a business honeypot to defend virtual machines (VMs) in Cloud Infrastructure. (ICI). Honeyed honeypots with Snort enable the detection of hidden security vulnerabilities and the prevention of internal intrusions or attackers exploiting them. According to the results of the experiment, the enterprise honeypot has been implemented effectively at ICI and is effective against security risks.

In 2020, Ahilan Appathurai, *et al*. [25] presented a collaborative active defensive technique between Honeypot and cloud platforms to identify and defend against future DDoS attacks interms of Internet of Things with instantaneous harmful traffic estimated at Terabytes. Using major simulation tools, the first stage of such a design and execution has been completed, and relevant sample results are presented in this report.

In 2021, Liu, Z., *et al.* [29] proposed a lightweight Privacy-Preserving Trust Evaluation (LPPTE) scheme which can efficiently balance privacy preservation and trust evaluation, with minimal overheads, for data fusion in cooperative vehicular safety systems. In simulated evaluations, the LPPTE technique outperforms the existing techniques in many ways, including its ability to improve fusion accuracy.

In 2021, Liu, Z., *et al* [30] proposed a Lightweight Trustworthy Message Exchange (LTME) system that combines cryptography and trust management technologies. The LTME scheme uses a centralised Ground Control Station (GCS) to securely deliver secret values to UAVs and periodically update their reputation levels. Compared to the existing systems, the proposed schemes serve robust functionality and have low computation and communication overheads.

In 2022, Guo, J., *et al* [31] proposed the intelligent cluster routing technique for UANETs. This module consists of a clustering component, a clustering adjustment component, and a routing component. The results indicate that ICRA may outperform its cutting-edge competitors in the context of clustering efficiency, topological stability, energy efficiency, and quality of service.

In 2020, Liu, Z., *et al.*,[32] developed a technique called BF-based private set intersection (PSI) that uses bloom filter (BF) technology. It is a revolutionary technique that can provide both exact trust management as well as strong conditional privacy preservation at the same time. Experimental analysis shows that the suggested scheme outperforms the existing schemes.

In 2019, Liu, Z., *et al* [33] proposed a novel trust cascading-based emergency message dissemination model (TCEMD) that integrates entity-oriented trust values with data-oriented trust evaluation. Based on simulations and studies performed in a typical highway environment, the proposed model performs much better than the existing models in several situations.

From the literature review, various techniques were studied but, these methods do not focus on zero-day attacks. In this paper, we focus on this problem by introducing a novel R2NET system that hybrids the RR framework with the honeynet system.

3. Proposed Method

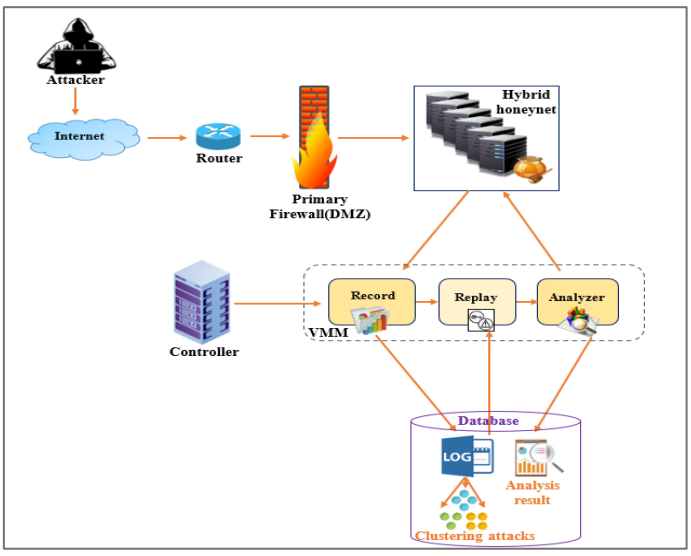


Fig. 1. Systematic representation of proposed framework.

R2NET is a honeynet system that uses the RR framework of the VM system. As shown in **Fig. 1** it comprises of four modules: Controller, Record Component, Replay Component, and Analyzer.

In particular, the honeynet system is made up of multiple honeypots that interact at both a low and high level. Using the Controller, users can control the honeypot system via VMM. When the Controller issues a record command, the Record Component begins operating. A honeypot collects all nondeterministic events and logs them into the storage system during its execution.

3.1 Hybrid Honeynet

The cloud is susceptible to both external and internal threats because of its wide range of uses and types of communications. The majority of attacks originate from the outside, so we must understand the attackers' information and intentions, as well as where they are coming from. To perform the necessary mitigation procedures, data such as the originating application vulnerabilities, operating system, IP address, type of service, and ports used must be known beforehand. For securing the system, IDS, IPS, Firewalls, cryptography, antivirus, and other technologies are available for implementation. It is impossible to look into how attackers gain access to the system, how they use it, or how the attack is carried out with these tools or methods. Although Honeypot and Honeynet deceive attackers into thinking they have access to the system's true assets, they have only entered a simulated system that logs their attack patterns. The seven categories that makeup Honeypot are application, scalability, resource level, source code accessibility, amount of engagement, and purpose. Here, we are considering the level of interaction because we are utilizing both high level and low levels of interaction.

The first step in setting up a honeypot system is to disguise it as a legitimate company server. False databases were set up and service ports, such as 80 (web service) and 110 (mail service), were opened. A honeypot system includes phishing materials, such as encrypted confidential data, to entice attackers into downloading and examining them [26-28]. Additionally, Vmtools and the MAC address of the virtual network card are removed from the honeypot's virtualization characteristics. With these cloaking techniques, the attacker will have a hard time determining the honeypot's legitimacy. The next step is to stay for the attacker to arrive and observe their behavior after deploying the honeypot environment.

1. In the Record component, packets are captured from the honeypot's sniffer and processed before being stored in the database.
2. The replay controller gives analyzers the ability to relive attack execution scenarios and an interface to start, pause, and stop the replay process. It also has a debugging environment like gdb.
3. The data analyzer examines and visualises the honeypot's harmful data. The results are then sent to the management via Web service. It plays a critical function in this system in recognising the features of attackers' behaviour. It also provides a significant reference for whether or not to keep the logs.

3.2 Record and Replay Framework

There are numerous non-determined factors that can affect the execution of a system. In honeypot operations, the Record Component is in charge of recording non-deterministic events. The component stores the non-deterministic events acquired by VMM in the buffer and then flushes them into the permanent storage system when the buffer is full. The Replay Component generates a Honeypot Replayer in the replay stage to replay the previous honeypot execution.

It sends non-deterministic events to Replayer via VMM by extracting them from log files. The sequence diagram for the suggested framework is shown in **Fig. 2**. According to logged data, the honeypot's execution will be rebuilt.

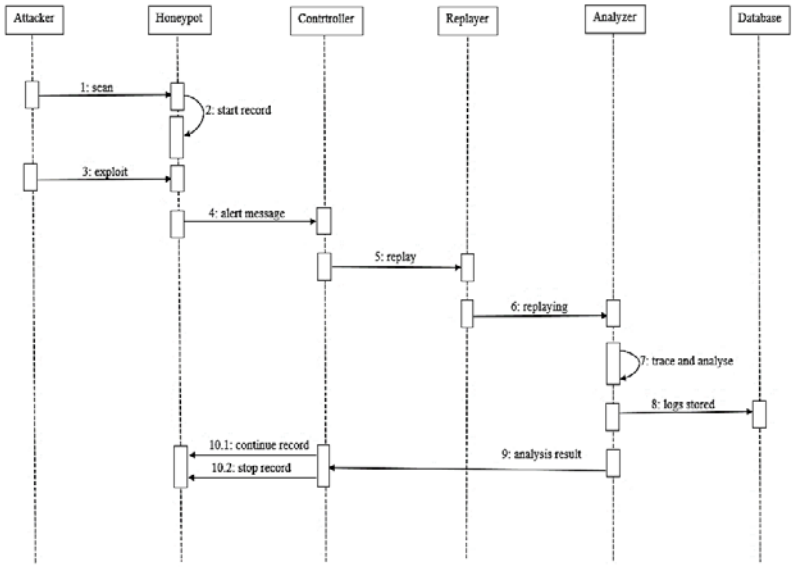


Fig. 2. Sequence diagram for the proposed framework.

3.3 Clustering

Due to the rapid advancement of network technology and the constant enhancement of technology, novel attack methods are appearing regularly. Connectivity analysis should be possible for intrusion detection systems using a variety of inputs. Fuzzy systems for intrusion detection have several key qualities, which are discussed below.

- Fuzzy systems' ability to integrate information from several sources
- Some types of invasions are difficult to detect
- The number of alarms that may occur as a result of an intrusion is often unknown.

In fuzzy clustering, objects are grouped based on statistical techniques. An object can belong to more than one cluster with varying degrees of membership, making objects in one cluster more likely to interact with those in another. The key advantage of clustering is its capacity to detect new attack patterns. Fuzzy C-Means Clustering (FCM) is the most widely used fuzzy clustering algorithm. A finite collection of provided samples is split into a set of fuzzy clusters based on a set of criteria. Most research topics in the cloud environment concentrate on high accuracy, despite other sequence issues like false alarms and response times. A hybrid honeypot that applies R2NET and the fuzzy c mean algorithm is suggested for detecting normal and attack behaviour. This work is characterized by a high level of detection accuracy.

Algorithm (1) : Training stage

Input : Number of samples chosen from the UNSW-NB 15 dataset, as well as the fuzziness parameter.

Output : Vector of two cluster center l= {l1, l2}

Steps

From the training samples, two cluster centers were randomly selected.
For each sample, calculate the membership matrix employing equation (2).
Using the membership matrix and equation, update the cluster centers (3).
Continue to rep until all criteria are met.
End

Algorithm (2) : Testing stage

Input : A two-cluster vector has been constructed from the UNSW-NB 15 dataset to select the number of testing samples.

Output : Various categories of samples were assigned according to attacker behavior.

Steps:

- Using the cluster center from the training step, calculate the membership matrix for every sample employing equation 2.
- Using the membership matrix and formula below, determine which cluster type corresponds to each sample.

$$If(m_{x1} > m_{x2}) then\ l_y(s_x) = 1$$
$$Else \quad l_y(s_x) = 2$$
$$End\ if$$
$$End$$

3.4 Dataset Description

UNSW-NB 15 data sets were generated using an IXIA PerfectStorm tool (IXIA PerfectStorm One Tool, 2014) in the Australian Centre for Cyber Security Cyber Range Lab, which generated synthetic modern attack behaviours from network traffic as well as real-world contemporary routine activities. In 2014, 100 GB of unprocessed network traffic has been recorded using tcpdump (tcpdump utility). The Bro-IDS and Argus tools were employed to extract the features, and 12 models were generated. The class label was used to extract 49 characteristics using these techniques in parallel processing. Four CSV files containing 2,540,044 records were created after the configured operations were completed. The datasets are divided into training and testing sets randomly in 80:20 proportion.

3.5 Fuzzy C-Means Algorithm (FCM)

The purpose of fuzzy clustering is to divide data into different clusters based on degree membership values and to group information that is identical to each other and distinct from data in other clusters. The proposed R2NET system is more efficient since this reduces the time it takes to access information. Soft clustering occurs when a data point is included in more than one cluster with varying degrees of FCM participation. The data is clustered according to a fuzzy membership function. Based on minimising objective y, fuzzy clustering

uses equations (1) to calculate. This equation (1) is derived on the basis of following object function minimization y.

$$O_r(M,L) = \sum_{x=1}^{d}\sum_{y=1}^{e} m_{xy}^{r} n_{xy}^{2}(s_x, l_y) \quad (1)$$

- r-real numbers in the domain
- e-number of clusters
- d-number of data samples
- m_x- membership degree which indicates the possibility that data sample $s_x \in y^{th}$ cluster
- l_y- center of cluster

Fuzzy clustering is achieved by repetitively modifying the cluster centre and fuzzy membership with equation 2.

$$m_{xy} = \frac{1}{\sum_{y=1}^{e}\left({}^{n_{xy}}/_{n_{xe}}\right)^{2/r-1}}, \qquad \forall x \quad (2)$$

$$l_y = \frac{\sum_{x=1}^{d} m_{xy}^{r} s_x}{\sum_{x=1}^{d} m_{xy}^{r}}, \qquad \forall x \quad (3)$$

Data samples that met these criteria and belonged to a given cluster had m_{xy} as the degree of membership:

$$\sum_{y=1}^{e} m_{xy} = 1 \qquad \forall e \quad (4)$$

$$\sum_{y=1}^{e} m_{xy} > 0 \qquad \forall x \quad (5)$$

There are two phases to the proposed FCM algorithm for analysing cyber attacker activity. Training occurs during phase one, during which a cluster center is identified. After the training phase, a second phase determines the cluster of newly generated samples using the cluster center result from the training phase. The proposed module's training stage was represented by an algorithm (1), while its testing stage was represented by an algorithm (2).

4. Results and Discussion

In this section, the suggested R2NET is evaluated with different measures. So that the system continues to record abnormal incoming network traffic, it is built to be self-updating.

4.1 Performance Analysis

The suggested framework is analyzed with the metrics for accuracy, False Positive Rate (FPR), F1-score, and specificity.

4.2 Accuracy

Accuracy is the system's ability to recognize traffic in both abnormal and typical situations. This is the percentage of traffic records that are correctly categorized out of the total number of records.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (6)$$

4.3 False Positive Rate (FPR)

As a result of the false positive ratio, the null hypothesis can be rejected incorrectly. The false positive rate is calculated using the following formula.

$$FPR = \frac{FP}{FP+TN} \tag{7}$$

The number of false positives, while the number of true negatives. The probability that a false alarm will be triggered, leading to a positive outcome while the genuine value is negative.

4.4 F1-score

F1 scores are calculated based on precision and recall using the harmonic mean. The harmonic mean is an alternative to the more commonly used arithmetic mean. Calculating an average rate using this method is often advantageous

$$F1 - score = \frac{TP}{TP+\frac{1}{2}(FP+FN)} \tag{8}$$

TP , FP, and FN represents true positives, false positives, and false negative.

4.5 Specificity

For each category, a model's specificity indicates how well it can forecast negative outcomes.

$$Specificity = \frac{(True\ Negative)}{(True\ Negative + False\ Positive)} \tag{9}$$

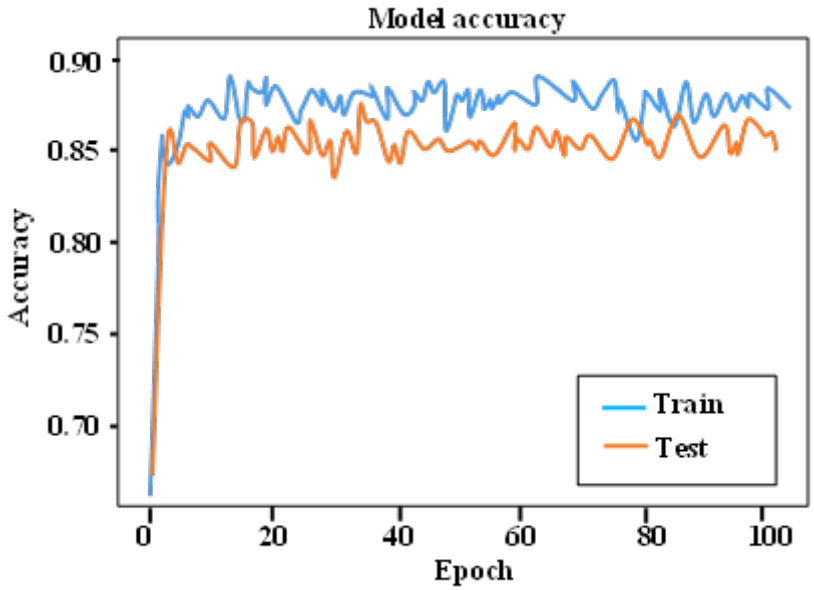


Fig. 3. Accuracy graph for testing and training of the FCM system.

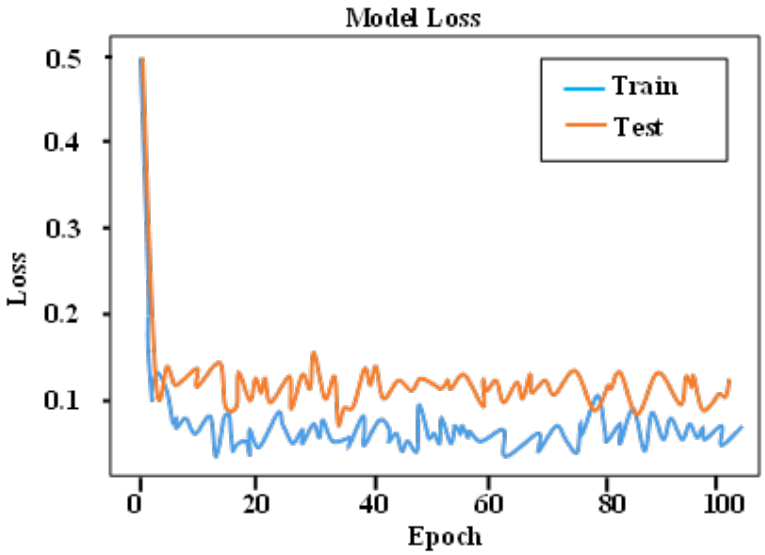


Fig. 4. Loss graph for the testing and training of FCM system.

Fig. 3 shows the accuracy graph for the testing and training of FCM system. The FCM achieves high accuracy in testing phase than the training phase. Thus, the performance of the suggested framework increases. **Fig. 4** shows the loss graph for testing and training of FCM which has less loss rate in testing than in the training phase which in turn automatically increases accuracy.

Table 1. Engagement times in RR framework vs stationarily implemented honeypots

RR framework in honeynet		**Stationarily implemented honeypot**	
Attacker IP	Time	Attacker IP	Time
172.16.238.41	5746 sec	20.45.26.133	346 sec
172.16.238.72	4258 sec	104.152.184.341	247 sec
172.16.238.59	3574sec	162.158.175.241	214 sec
172.16.238.24	3154sec	106.204.451.214	165 sec
172.16.238.56	2456sec	106.204.451.512	114 sec
172.16.238.86	2365sec	106.144.451.184	66 sec
172.16.238.51	1945sec	45.135.142.45	50 sec

According to **Table 1**, the time it takes for the proposed R2NET framework to engage an attacker is longer than a honeypot placed stationarily. As a result, stationarily implemented honeypots attracted less number of attackers and consumed more resources than R2NET solutions. In conclusion, honeynet delivered by the RR framework reduces attacker engagement time and saves resources, thus enhancing honeypot deployment efficiency. They are classified as behavioural honeypots since they are only used when an attacker is present.

Fig. 5 shows the results of detecting speed. While using the hybrid honeynet system, detection mode processing speed is significantly faster than campus network traffic (the

maximum and average traffic log per second are respectively 5,520 and 3912 records/s). This system can even handle large amounts of traffic.

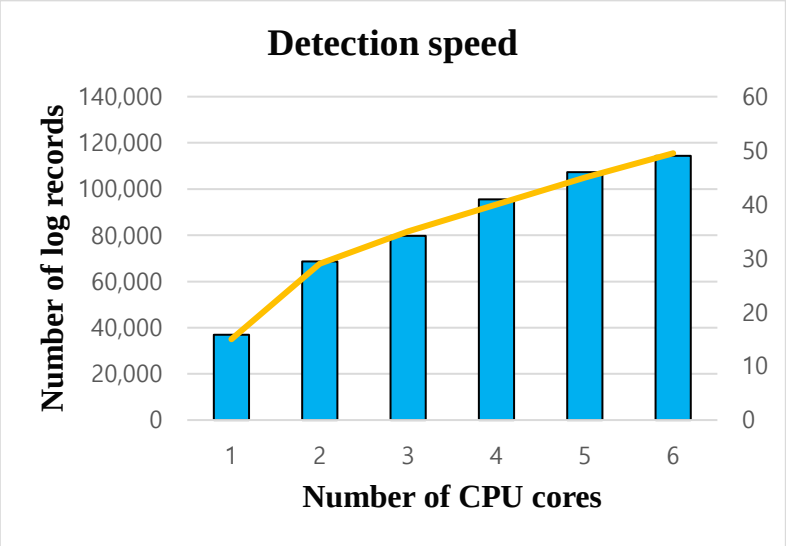


Fig. 5. Detection speed of the proposed method.

4.6 Comparative Analysis

A comparison of existing approaches was conducted to demonstrate that the suggested scheme produces more effective results. The performance is based on the F1 score, specificity, precision, recall, and accuracy. The accuracy rate is obtained by the R2NET is more efficient than the existing models. This analysis compares the proposed framework with the three major deep learning techniques.

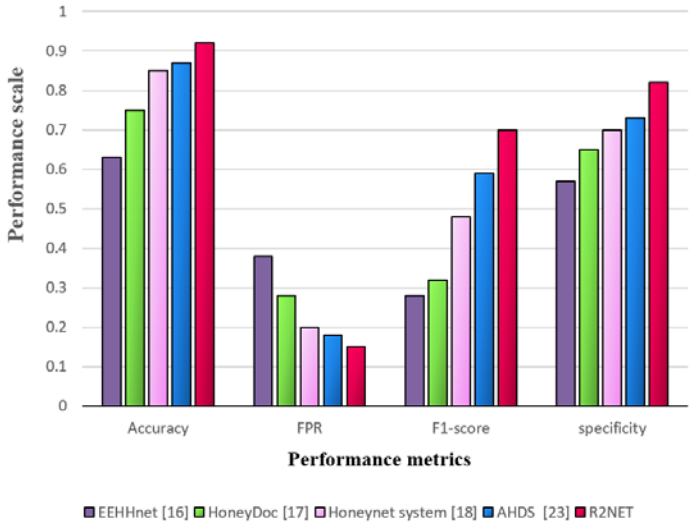
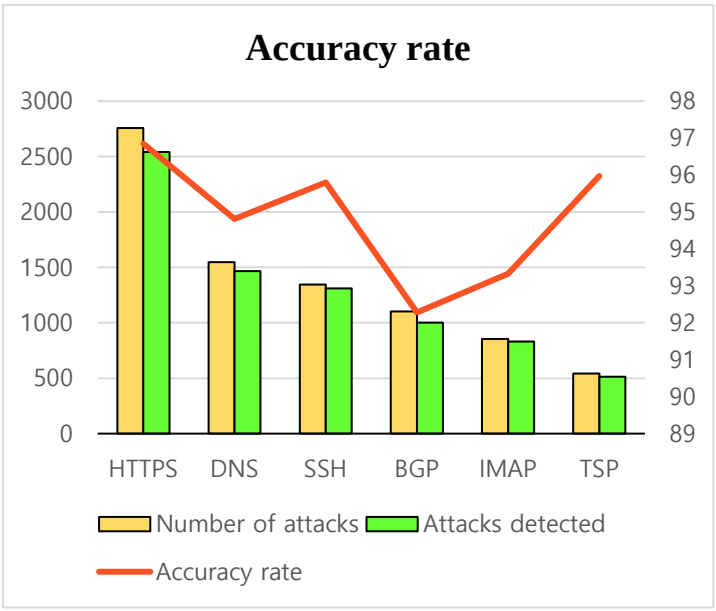
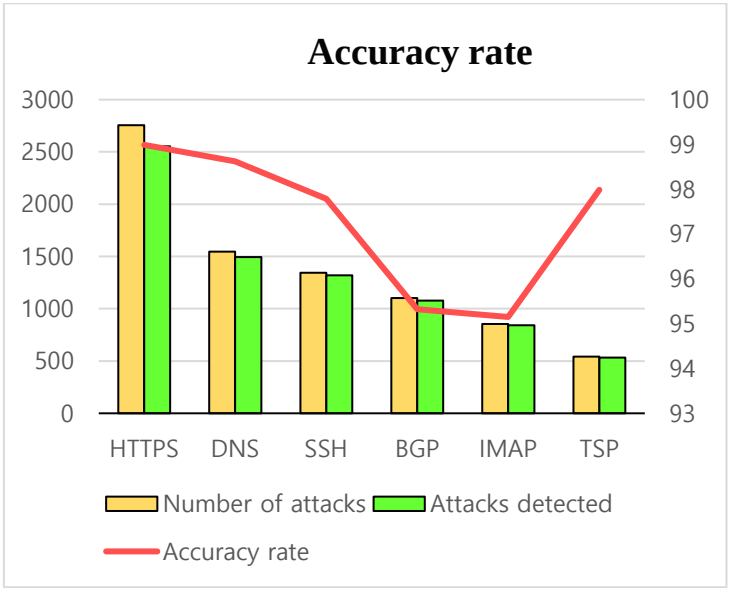


Fig. 6. Performance metrics.

Fig. 6 represents the comparison of performance metrics including accuracy, FPR F1-score, and Specificity of the suggested framework with the existing systems such as EEHH net, HoneyDoc, honeynet system, and AHDS. Fig. clearly shows that the proposed method achieves higher accuracy, F-score, and specificity and reduces the false positive rate.



(i)



(ii)

Fig. 7. (i) Accuracy rate of honeynet without RR framework. (ii) Accuracy rate of honeynet with RR framework.

Fig. 7 (i) and **7 (ii)** depict the number of attacks, detected attacks, efficiency rates in a cloud network, port numbers, and considering an attack. **Fig. 7(i)** represents the data that can be

found without utilizing the RR framework, while **Fig. 7 (ii)** illustrates the results that can be found with it. These graphs show that when the RR framework was active, higher success rates in detecting attacks were obtained for each protocol. Due to the problem of identifying zero-day attacks in real-time, performance can suffer, particularly in anomaly-based systems. There was an improvement of 1.34 percent to 3.81 percent in this study. Rates vary for different periods. R2NET is capable of recording, replaying, analyzing, and filtering data.

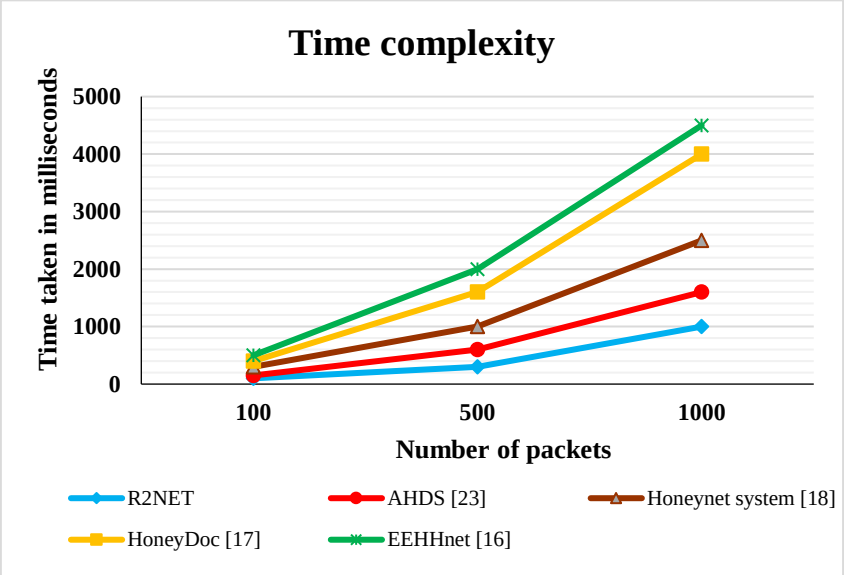


Fig. 8. Proposed system the time complexity.

Fig. 8 compares the proposed R2NET system to various techniques based on their time complexity. When compared to alternative systems for varying numbers of packets, R2NET requires a very short time. Other methods take longer to analyze and detect an intrusion for a given number of packets than the suggested system.

4.7 Evaluation of UNSW-NB15

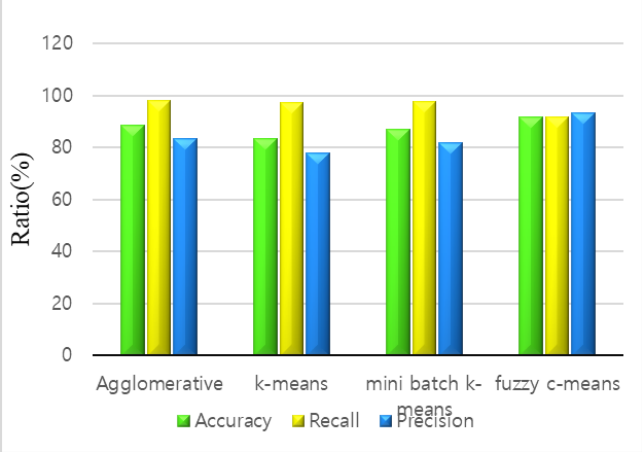


Fig. 9. Comparison of UNSW-NB15 test results.

Fuzzy c means to achieve 92 percent accuracy, recall, and precision in the accuracy, recall, and precision evaluation parameters. Experimental results are compared with the other three classic approaches. Comparison of UNSW-NB15 test results shown in **Fig. 9**. Compared to our proposed R2NET method, these three models have a higher recall (about 97%), lower accuracy (around 83–87%), and poorer precision (around 78–82%) detection pattern. In addition, of all the models evaluated, our proposed technique had the highest accuracy (91.8%) and precision (93.2%).

Table 2. Evaluation accuracy of FCM

Logs	Evaluation accuracy			
	Agglomerative	**K-means**	**Mini-batch k-means**	**FCM**
1	85.43	88.23	89.43	98.21
2	84.65	86.09	88.76	97.09
3	80.72	83.23	87.43	96.33
5	79.22	82.12	86.55	95.43
7	78.86	81.06	85.72	93.24
9	78.47	80.72	84.86	91.43

Table 2 shows the results obtained in terms of overall accuracy. From **Table 2** the traditional clustering algorithms namely Agglomerative, k-means and mini batch k-means obtains less accuracy compared to the FCM. FCM preserves the high accuracy range of 98.12%. Thus, it is seen that FCM outperforms other algorithms.

5. Conclusion

This paper presents the R2NET honeypot system. Honeypots can be algorithmically reproduced by collaborating with VM's recording and replay capabilities. It will be possible to perform an even more precise analysis with this advancement in the system by delaying the analysis until after the replay. The logs stored in the database are then clustered in terms of attacks accordingly. So, the accessing time for analyzing the attack may be reduced which in turn increases the efficiency of the proposed framework. In real-world application cases, R2NET has proven effective and versatile. Virtualization and network protection are combined in this research, resulting in a new paradigm for defence systems. The proposed R2NET framework is compared with existing methods such as EEHH net, HoneyDoc, Honeynet system, and AHDS. The proposed system achieves 7.60%, 9.78%%, 18.47%, and 31.52% more accuracy than EEHH net, HoneyDoc, Honeynet system, and AHDS methods. Future investigations of attacks will be conducted, and possible algorithmic responses will be implemented. Furthermore, we also suggest implementing the proposed techniques on real-world platforms for verification of the presented results.

Acknowledgment

The Author with a deep sense of gratitude would thank the supervisor for his guidance and constant support rendered during this research.

References

[1] B. T. Devi, S. Shitharth and M. A. Jabbar, "An Appraisal over Intrusion Detection systems in cloud computing security attacks," in *Proc. of 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, IEEE, pp. 722-727, 2020. Article (CrossRef Link)

[2] V. Mahajan, and S. K. Peddoju, "Integration of network intrusion detection systems and honeypot networks for cloud security," in *Proc. of 2017 International Conference on Computing, Communication and Automation (ICCCA)*, IEEE, pp. 829-834, 2017. Article (CrossRef Link)

[3] M. Baykara, and R. Das, "A novel honeypot-based security approach for real-time intrusion detection and prevention systems," *Journal of Information Security and Applications*, vol. 41, pp. 103-116, 2018. Article (CrossRef Link)

[4] N. S. Safa, C. Maple, T. Watson and R. Von Solms, "Motivation and opportunity-based model to reduce information security insider threats in organisations," *Journal of information security and applications*, vol. 40, pp. 247-257, 2018. Article (CrossRef Link)

[5] S. Sağiroğlu, E. S. R. A. Yolacan and U. Yavanoğlu, "Designing and developing an intelligent intrusion detection system," *Journal of the Faculty of Engineering and Architecture of Gazi University*, vol. 26, no. 2, pp.325-340, 2011.

[6] S. S. Chakkaravarthy, D. Sangeetha, M. V. Cruz, V. Vaidehi, and B. Raman, "Design of intrusion detection honeypot using social leopard algorithm to detect IoT ransomware attacks," *IEEE Access*, vol. 8, pp. 169944-169956, 2020. Article (CrossRef Link)

[7] A. Mairh, D. Barik, K. Verma, and D. Jena, "Honeypot in network security: a survey," in *Proc. of the 2011 international conference on communication, computing & security*, pp. 600-605, 2011. Article (CrossRef Link)

[8] G. Kumar, R. Saha, M. Singh, and M. K. Rai, "Optimized packet filtering honeypot with snooping agents in intrusion detection system for WLAN," *International Journal of Information Security and Privacy (IJISP)*, vol. 12, no. 1, pp. 53-62, 2018. Article (CrossRef Link)

[9] A. D. Akshay, A. Bhushan, N. Anand, and R. Khemka, "HONEYPOT: Intrusion detection system," *International Journal of Education, Science, Technology, and Engineering*, vol. 3, no. 1, pp. 13-18, 2020. Article (CrossRef Link)

[10] S. Krishnaveni, S. Sivamohan, S. S. Sridhar, and S. Prabakaran, "Efficient feature selection and classification through ensemble method for network intrusion detection on cloud computing," *Cluster Computing*, vol. 24, no. 3, pp. 1761-1779, 2021. Article (CrossRef Link)

[11] A. Pashaei, M. E. E. Akbari, M. Z. Lighvan, and A. Charmin, "Machine Learning-Based Early Intrusion Detection System in Industrial LAN Networks Using Honeypots," 2021. Article (CrossRef Link)

[12] K. Ramakrishnan, P. Gokul, and R. Nigam, "Pandora: An IOT based Intrusion Detection Honeypot with Real-time Monitoring," in *Proc. of 2021 International Conference on Forensics, Analytics, Big Data, Security (FABS)*, IEEE, vol. 1, pp. 1-7, 2021. Article (CrossRef Link)

[13] A. Rohit, B. N. Jaswanth and C. S. Reddy, "Intrusion Detection System Using Honey Pot," 2021.

[14] S. Shyla and V. Bhatnagar, "The Geo-Spatial Distribution of Targeted Attacks sources using Honeypot Networks," in *Proc. of 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, IEEE, pp. 600-604, 2021. Article (CrossRef Link)

[15] P. Wang and H. D'Cruze, "Honeypots and knowledge for network defense," *Issues in Information Systems*, vol. 22, no. 3, pp. 241-254, 2021. Article (CrossRef Link)

[16] N. K. Bao, S. W. Ahn and M. Park, "An elastic-hybrid honeynet for cloud environment," *CSEIT, NCS, SPM, NeTCoM*, pp. 117127, 2018.

[17] W. Fan, Z. Du, M. Smith-Creasey and D. Fernandez, "Honeydoc: An efficient honeypot architecture enabling all-round design," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 3, pp. 683-697, 2019. Article (CrossRef Link)

[18] M. Marydas and J. N. Varsha Priyah, "A cloud based honeynet system for attack detection using machine learning techniques," *International Research Journal of Engineering and Technology (IRJET)*, vol. 6, no. 7, pp. 330-335, 2019.

[19] M. A. Saxena, G. Ubnare, and A. Dubey, "Virtual Public Cloud Model in Honeypot for Data Security: A New Technique," in *Proc. of the 2019 5th International Conference on Computing and Artificial Intelligence*, pp. 66-71, 2019. Article (CrossRef Link)
[20] S. K. Sood and K. D. Singh, "Hmm-based secure framework for optical fog devices in the optical fog/cloud network," *Journal of Optical Communications*, 2019. Article (CrossRef Link)
[21] H. Al-Mohannadi, I. Awan, and J. Al Hamar, "Analysis of adversary activities using cloud-based web services to enhance cyber threat intelligence," *Service Oriented Computing and Applications*, vol. 14, no. 3, pp. 175-187, 2020. Article (CrossRef Link)
[22] E. M. Kandoussi, M. Hanini, I. El Mir, and A. Haqiq, "Toward an integrated dynamic defense system for strategic detecting attacks in cloud networks using stochastic game," *Telecommunication Systems*, vol. 73, no. 3, pp. 397-417, 2020. Article (CrossRef Link)
[23] T. Kong, L. Wang, D. Ma, Z. Xu, Q. Yang, Z. Lu, and Y. Lu, "Automated Honeynet Deployment Strategy for Active Defense in Container-based Cloud," in *Proc. of 2020 IEEE 22nd International Conference on High Performance Computing and Communications; IEEE 18th International Conference on Smart City; IEEE 6th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, IEEE, pp. 483-490, 2020. Article (CrossRef Link)
[24] K. D. Singh, "Securing of Cloud Infrastructure using Enterprise Honeypot," in *Proc. of 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, IEEE, pp. 1388-1393, 2021. Article (CrossRef Link)
[25] Ahilan Appathurai, Revathi Sundarasekar, Chandrasekaran Raja, E. John Alex, C. Anna Palagan, and A. Nithya, "An efficient optimal neural network-based moving vehicle detection in traffic video surveillance system," *Circuits, Systems, and Signal Processing*, vol. 39, no. 2, pp. 734-756, 2020. Article (CrossRef Link)
[26] R. Surendiran, "A Secure Command Based Approach to find Stolen Mobiles," *Research Review International Journal of Multidisciplinary*, vol. 3, no. 10, pp. 454–456, 2018.
[27] J. X. Huang, S. Zhou, N. Savage, and W. Zhang, "A distributed cloud Honeypot architecture," in *Proc. of 2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)*, IEEE, pp. 1176-1181, 2021. Article (CrossRef Link)
[28] G. Susan Shiny and B. Muthu Kumar, "E2IA-HWSN: Energy Efficient Dual Intelligent Agents based Data Gathering and Emergency Event Delivery in Heterogeneous WSN Enabled IoT," *Wireless Personal Communications*, vol. 122, no. 1, pp. 379-408, 2022. Article (CrossRef Link)
[29] Z. Liu, J. Ma, J. Weng, F. Huang, Y. Wu, L. Wei, and Y. Li, "LPPTE: A lightweight privacy-preserving trust evaluation scheme for facilitating distributed data fusion in cooperative vehicular safety applications," *Information Fusion*, vol.73, pp.144-156, 2021. Article (CrossRef Link)
[30] Z. Liu, J. Guo, F. Huang, D. Cai, Y. Wu, X. Chen and K.K. Igorevich, "Lightweight Trustworthy Message Exchange in Unmanned Aerial Vehicle Networks," *IEEE Transactions on Intelligent Transportation Systems*, pp.1-14, 2021. Article (CrossRef Link)
[31] J. Guo, H. Gao, Z. Liu, F. Huang, J. Zhang, X. Li and J. Ma, "ICRA: An Intelligent Clustering Routing Approach for UAV Ad Hoc Networks," *IEEE Transactions on Intelligent Transportation Systems*, 1-14, 2022. Article (CrossRef Link)
[32] Z. Liu, F. Huang, J. Weng, K. Cao, Y. Miao, J. Guo, and Y. Wu, "BTMPP: balancing trust management and privacy preservation for emergency message dissemination in vehicular networks," *IEEE Internet of Things Journal*, vol.8, no.7, pp.5386-5407, 2021. Article (CrossRef Link)
[33] Z. Liu, J. Weng, J. Ma, J. Guo, B. Feng, Z. Jiang, and K. Wei, "TCEMD: A trust cascading-based emergency message dissemination model in VANETs," *IEEE Internet of Things Journal*, vol.7, no.5, pp.4028-4048, 2020. Article (CrossRef Link)

M. R. Amal was born in Trivandrum, Kerala, India in 1993. He received Bachelor degree in Computer Science and Engineering from Kerala University, Trivandrum in 2014. He received his Master degree in Computer Science and Engineering from Kerala Technological University, kerala, India in 2017. Currently pursuing Ph.D. in Computer Science and Engineering with Noorul Islam Centre for Higher Education, Kumaracoil, Thuckeley, Kanyakumari district, Tamilnadu. He is working as an Assistant Professor in the Department of Computer Science, St. Albert's College (Autonomous), Kochi, Kerala. He had published number of papers in various National, International journals & conferences. His research areas include Big Data processing and Cloud Security.

P. Venkadesh was born in Nagercoil, TamilNadu, India in 1980.He studied Computer Science & Engineering at C.S. I Institute of Technology, Thovalai, TamilNadu, India. He received Bachelor degree from M. S University, Tirunelveli in 2001.He received his Master degree from Sathyabama University, Chennai, TamilNadu, India in 2007. He Completed his Ph.D. at Noorul Islam Centre for Higher Education in 2017 under the area of Network Security. Currently, he is working as an Assistant Professor in the Department of Computer Science & Engineering at Noorul Islam Centre for Higher Education, Noorul Islam University, Kumaracoil, TamilNadu, India. He had published and presented a no. of papers in National, International Conferences and also in various referred Journals. His research area includes Network Security, Image Processing, Wireless Communications and Cloud Computing.