Reinforcement Learning for Detection and Prevention of DDoS Attacks in Cloud Environment

Khaled Omer Basulaim *k.basulaim@moheye.net*
Faculty of Engineering
University of Aden
Aden, Yemen

Hanan Mohammed AL-Amoudi *hananmohd231@gmail.com*
Faculty of Engineering
University of Aden
Aden, Yemen

Abstract

Cloud computing is regarded as the one of the key technologies today as it provides the resources based on the on-demand availability of the users. Even though it provides reliable services, security is one of the major concerns. One of the major security threats in the cloud computing environment is Distributed Denial of Service (DDoS) attacks which makes the resources unavailable to the end users by exploiting the entities through continuous requests in distributed locations. The proposed work aims to solve the prior problems by proposing Reinforcement Learning based DDoS in Cloud (RL-DDoS Cloud). The proposed work adopts RL algorithms for network adaptively which also satisfy the QoS of the users. In order to provide framework against DDoS attacks, this work performs three stages such as DDoS prevention, DDoS Detection, and Risk Aware VM Isolation. In first stage, the proposed work checks the legitimacy of the users by multi factor authentication method which adopts SHA-512 algorithm. In second stage, the user data packets are selected and analyzed using fuzzy VIKOR algorithm which ensures security against inside attackers. Finally, the third stage provides secure VM migration and isolation using Soft Actor Critic algorithm by considering VM status and optimal VM selection metrics. The proposed work is simulated using Cloudsim simulation tool and evaluated with several validation metrics. The validation results shows that the proposed work outperforms better than the existing works.

Keywords: Cloud Computing, Reinforcement Learning, DDoS Attack.

1. INTRODUCTION

Cloud computing is one of the major paradigms in recent days as it provides resilient services to the users through internet based on “pay as you use basis” (P. Sareenet, al 2013). The cloud environment consists of set of physical machines, virtual machines, servers which provides services to the end users include storage, runtime environment, testing environment, etc.., which are partially completely or fully managed by the end user rather cloud services providers (CSPs) takes full charge on management (N. Joshi et al 2019), (V. Chang et al 2018).

Generally, cloud computing provides three types of services such as Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS) (M. Walterbusch et al 2018) (O. Yigitbasioglu et al 2014).The SaaS is completely managed by the CSPs whom help to connect the cloud users with cloud-based applications over internet (B. Martens et al 2015). The PaaS is not completely managed by the CSPs rather they manage virtualization services, runtime, operating system, networking, storage, servers, and middle ware, and cloud users are responsible to manage application and data (I. Baltatescu et al 2014). Similarly, to the PaaS, the IaaS was also not completely managed by the CSPs in contrast to PaaS they only manage

servers, storage, virtualization services, and network (R. Bhoyar et al 2013). However, the other services such as application, data, runtime, middleware, and operating system were managed by the end users.

These services enable the cloud computing to ensure reliability and resiliency. Even though the cloud computing ensures reliability and resiliency, The security in cloud computing was a major concern. The security breaches in the cloud computing environment makes them to degrade its nature and also downs the QoS of the users (A. Khalid et al 2013), (J. Askhoj2010).

DDoS is the one of the major attacks in the cloud computing environment in which the attacker makes the environment gets overloaded through continuous flood of request. These types of attacks are managed by single organization which utilizes bots to pose attacks in distributed manner by many locations (R.K. Deka et al 2017), (A. Chaturvedi et al 2018). The DDoS was firstly identified in the year 1999 (Joosten & L.J et al2017). However, in later years many of the large companies such yahoo, eBay, Amazon was affected the DDoS attacks. The twitter, Facebook services were also affected by the DDoS attacks in the year of 2009 (D. Ameyedet al 2015), (A. Abhishta et al 2015).

In the year of 2010 parliament of Australia was affected, and election commission website of South Korea was also affected in the same year. More specifically, the powerful countries in the world such as Russian and American government websites were also attacked by the DDoS attacks in the year of 2011 and 2012 . Many of the works focusing on detecting and mitigating the effect of DDoS through several techniques. The existing works includes trust-based method, game theory models are proposed by the researchers, but the adaptability to the real time environment was lacked on their works (C.Culnaneet al 2017).

Some work adopts artificial intelligence techniques for DDoS detection. Almost many of the researchers adopt machine learning algorithms, optimization algorithms, and threshold-based methods for DDoS detection (M. Alkasassbehet al 2017). However, machine learning algorithms are limits with less detection accuracy and error prone, optimization algorithms may fall on local optima while threshold-based method faces severe complexity in the real time environment V. (Behzadanet al 2018). To overcome these above issues, some of the existing works utilizes RL algorithms for DDoS Detection. However, the prior adopted RL algorithms was off-policy algorithm and are works in discrete time space which leads to high error and less adaptability (A. K. Lambaet al 2018). Even though the existing works provides many frameworks still the complete solution to the DDoS detection in terms of adaptability and security was not yet provided. The proposed work aims to solve the existing issues in the DDoS detection by proposing end-to-end strategy which also satisfies the QoS and SLA of the CSPs as well as the end users. To this end some of the research questions also provided below,

- How can we attain tradeoff among throughput and DDOS attack detection accuracy
- How can we manage the scalability issues in the cloud environment when detecting DDOS attacks

1.1 Motivations and Objectives

The major aim of the research work is to secure the cloud environment using reinforcement learning approach. Apart from that, this research also addresses the existing issues in terms of improper resource allocation, limited security, etc..,. To be more specific, the significant problems encountered in the existing research works are listed as below,

- Inadequate Security Approaches: The existing work performs security to the cloud environment against DDoS either externally or internally. The internal process includes DDoS detection using several methods, and external process includes authentication. However, the attacker might posses' attacks in both perspectives which was not handled by them that leads to high security breaches.

- Nonexistence of VM Isolation: The isolation of affected VM reduce the scalability issue and security issues in the cloud environment. The nonexistence of VM isolation makes the VM to overload. However, almost all the existing work considers only DDoS detection part and without considering mitigation which makes with highly vulnerable to resource exploitation of whole cloud environment.
- Improper Resource Allocation: Proper resource allocation in cloud environment improves the QoS. While most of the existing works performs improper resource allocation by considering only limited metrics which leads to poor QoS in terms of high energy consumption, high load, etc.., which makes the SLA to be violated in the cloud environment.

The above research problems motivated us to propose an efficient research work with the objective of detecting, preventing, and mitigating the DDoS attacks in cloud environment using reinforcement learning algorithms. Some of the sub-objectives of this research work are provided below,

- To enhance the DDoS detection accuracy in the cloud environment by performing effective DDoS detection in which the feature selection and data packet classification are taken place by considering several packet related features of the users.
- To improve the network scalability and minimize the impact of overloading, we perform risk aware VM isolation in case of probability at no risk and probability at risk respectively.
- To reduce the rate of illegitimacy and unwanted traffic, we perform authentication as a preventive measure by considering several user related metrics which reduces the attack impact at future.

1.2 Research Contributions

This approach mainly focusing on providing security to the cloud environment through three stages. The major contributions of the proposed work are provided below,

- A secure DDoS detection prevention technique is enabled by adopting multi factor authentication using SHA-512 algorithm based on multiple factors such as user ID, password, and random number. Based in these metrics, the proposed work verifies the user legitimacy and mitigate the unwanted network traffics. The adoption of multi factor authentication drastically reduces the vulnerability of insider attacks.
- The DDoS detection method is done by utilizing fuzzy VIKOR algorithm which selects the optimal packet features such as Source Destination IP Address, IP Address, Destination Port, Source Port, Protocol Type and Packet Size. The packet features are analyzed and classify into two classes as normal and malicious. This stage reduces the compromization attacks.
- The risk aware VM migration and isolation is done to reduce the manipulation of VMs by high potential outsiders and insiders using SAC which considers metrics such as VM states, CPU, storage, etc..,. This stage enables high QoS to the users without compromising the SLA requirements

1.3 Paper Organization

The remaining of the work is organized as follows, section II provides the background knowledge of the proposed methods and algorithm, section III explains the literature survey of the existing works which also shows the existing research gaps, section IV illustrates the proposed methodology are presented in detail with suitable mathematical equations, pseudocode, and diagram. Section V gives the experimental results in which the simulation setup, comparative analysis, and research summary are given, and section VI concludes the proposed work with possible future directions

2. BACKGROUND

This section provides the basic preliminary knowledge of the proposed method in which the any reader can understand our concept better. This section provides the information of reinforcement learning, and DDoS in cloud environment. The detailed explanation is provided below.

2.1 Reinforcement Learning

Reinforcement Learning (RL) is regarded as the class of machine learning algorithm in which the RL algorithm allows the agents to learn the environment and provides the appropriate reward to them (Botvinick, et al 2019). The Markov decision process (MDP) is typically used for modeling of any RL algorithms. As RL to be modelled with MDP, it can be well suited for real time and dynamic scenarios(Levine, S et al 2021), (Zhang, K et al 2019). It can also provide solution to the uncertain environments by determine the transition of sates through its probabilistic nature. Generally, MDP can be presented by considering tuples as $[St, Ac, \Pr(.|St, Ac), U(.|St, Ac)]$. Including headings, figures, tables and references. Manuscripts with poor or no typesetting are not preliminary approved and consider for review.

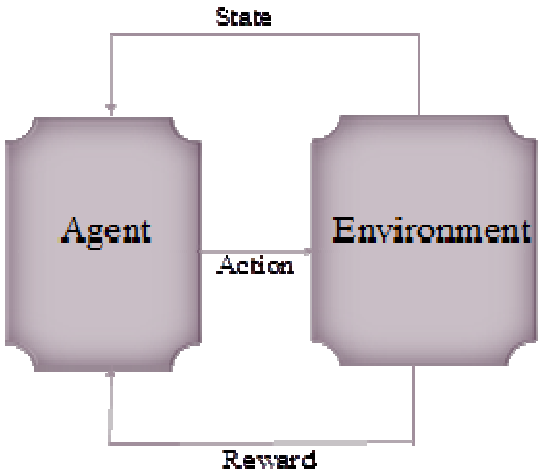

FIGURE 1: Illustrates the general flow of an RL process.

The St represents the state of the environment which can be further denoted as $St \in S$, Ac denotes the action space which can be further represented as $Ac \in Ac(St)$ in which the $Ac(St)$ denotes the actions within the given state space. $\Pr(.|St, Ac)$ denotes the state transition probability as $S_{t+1} \sim \Pr(.|St, Ac)$, and $U(.|St, Ac)$ denotes the reward probability as $Re(St, Ac) \sim U(.|St, Ac)$. The agent in the RL efforts to maximize their reward function based on the optimal actions which can be formulated as,

$$Re_{S,S'}^{Ac} = Max\{re_{t+1}|St = S, Ac = A, S_{t+1} = S'\} \quad (1)$$

The probability of state transitions in the environment gives the next state S_{t+1} of the agent based on the state and action (St, Ac) chose previously which can be formulated as,

$$Pr_{S,S'}^{Ac} = Pr\{S_{t+1} = S'|St = S, Ac = A\} \quad (2)$$

From the above equations (1) and (2), the rewards are provided for the transition states which was managed by the reward probability function $U(.|St, Ac)$. The process of reward maximization will be continued until the agent provides the error free solution. Fig 1 represents the general flow diagram of the RL algorithm.

2.2 DDOS in Cloud Environment

Distributed Denial of Service (DDoS) is one of the major attacks in the networking domain. The attacker efforts to overload the target resource by sending the unlimited bag of requests and make the resource unavailable to the victim (Bhardwaj et al 2021), (Dong, S et al 2019). Initially, the attacker manipulates the victim entities such as host, servers, virtual machine, agents, etc., with them, the attacker obscured the victim network connectivity leads to scalability issues. More specifically, the DDoS attacks may target for the victim resources such as virtual servers, CPU, storage servers, etc..,. In case of cloud environment, the attacker manipulates the virtual servers to weaken their connectivity. Typically, any DDoS attacker attacks the victim resource based on the two factors which are provided below:

- Attacker tries to starve the network by deplete the resource bandwidth of the target system.
- On the other hand, any attacker may find any infections in the software implementation to eat up the target resources. The figure (2) represent the DDoS attacker networks in detail.

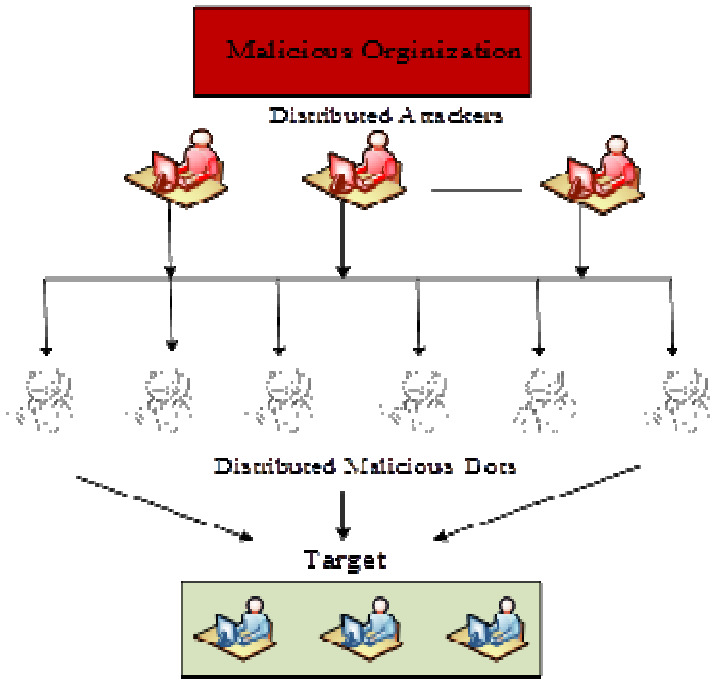

FIGURE2: DDoS Network.

3. LITERATURE REVIEW

This section provides the vast literature gap of the existing works in detail. Further, for better understand to the reader we also list the existing works in form of table I. The existing gaps are provided below.

O. Abdel. Wahab, J. Bentahar, H. Otrok, A. Mourad. Proposed a robust load circulation method for VMs in cloud environment to detect DDoS attacks. This work considers the existing issue of poor efficiency in attack detection. The prior issues in the existing works were addressed in this work by proposing two steps method named trust-based method and game theoretical approach called maximin and minimax game model. The relationship among the cloud systems and VMs

are evaluated under trust-based framework in terms of trust scores. Further, to achieve the robust DDoS detection with limited resources, the game theory model is proposed in which the strategy of the hypervisor was maximized to reduce the attacker strategy.

X. Chai, et al. enables Q-learning algorithm and MTD for DDoS attack detection in cloud environment. The existing issue of high time consumption for DDoS detection was addressed by proposing Q-learning and MTD. This work ensures the fair trade off among the security and cost constraints in which the cloud users could utilize the optimal strategy to mitigate the malicious activities. In addition, the dynamic network natures were adopted by Q-learning algorithm which improves the network adaptability. On the other side, the security among the users was provided by MTD strategy which increases the attacker's cost.

K. A. Simpson, S. Rogers, & D. P. Pezaros proposed an extenuation method for DDoS in cloud environment using direct control reinforcement learning. The state-of-the-art issues in terms of lack of network adaptability and high time consumption was addressed by proposing per host basis reinforcement learning method. The proposed method composed of two sub-steps named prompt action and coarseness action. In prompt action, every agent in the environment lessly involves with network traffic while in coarseness action the agent in the environment completely manipulates and assesses every flow of the host to mitigate the DDoS attacks in the cloud environment.

K. J. Singh, K. Thongam, & T. De using showy events by adopting fuzzy logic and genetic algorithms that was detected DDoS attacks. The prior works shows less effectiveness for detecting showy events and DDoS attacks which was overcome by proposing fuzzy logic and genetic algorithms. The fuzzy logic model taken input as many parameters such as uniqueness probability of internet protocol address, packet arrival time, hypertext transfer protocol unavailability. Here, the genetic algorithm was utilized for tuning the fuzzy parameters for providing optimal results. Finally, the output was given as normal or malicious traffic.

H.S. Mondal, M.T. Hasan, M.B. Hossain, M.E., Rahaman, &, R. Hasan was proposed the fuzzy logic-based DDoS attack detection in the cloud environment. The prior issues in terms of lack of cognitive behaviors were corrected in this approach by proposing fuzzy inference system. The proposed method acquires three inputs to the fuzzy inference system such as arrival rate of packet, IP from the source, and address of the port. The input was employed for fuzzification. Once fuzzified, the inference engine compared the inputs with already generated rules and defuzzified. Any input that deviates from the already generated rules was marked as malicious else normal.

M.S & P. Viswanathan proposed an energy constrained VM selection for cloud environment. The state of the art of frequent migration and inefficient load balancing was addressed by proposing consumption factor based VM selection. The proposed method evaluates the need of the resource for classification of task, VM migration, and load balancing. For reducing the energy consumption and utilization of resource rate Cramer Rao lower bound model was used, and several metrics was taken such as node state, and utilization of CPU rate for optimal VM migration.

S. Sambangi, & L. Gondiwas detected the distributed denial of service attacks by utilizing machine learning method The prior issues of high computation complexity were addressed by proposing machine learning based method. Initially, the traffics in the network analyzed and stored as log file. The optimal features were selected from the seized traffics. With the help of multiple linear regression, the normal and DDoS traffic was detected. This work adopts CICIDS dataset for machine learning model training and achieved better results.

S. Vetha, K. Devi proposed a DDoS prevention method using hypervisor trust mechanism. The existing work does not consider the resources in the cloud thereby leads to poor detection system. The proposed work leverages the existing disadvantage by hypervisor trust method in

which the visitant VMs are communicated in trusted manner through hypervisor. Further to reduce the attacker capability, the maximin game was proposed in which the hypervisor tries to minimize the attacker objective by maximizing defending strategy. Finally, the attacker patterns are provided to Tiniest square support vector machine for classified as normal and malicious traffics.

A. Sahi, D. Lai, Y. Li, M. Diykh proposed distributed denial of service transmission control protocol flooding attacks detection. The state-of-the-art works faces with high time consumption leading to resource hijacking. The proposed work mitigates those issues by performing detection and prevention method for distributed denial of service transmission control protocol flooding attacks. This work consists of two stages named detection and prevention stage. At detection stage, the traffics are seized such as time of arrival, source of arrival, length of the packet. Based on the features, the packets are classified as normal and malicious. Finally, at prevention stage, the corresponding sender will be discarded from the network and the pattern of attackers was stored in the blocklist. A prevention method for DDoS attack detection in cloud environment was done in proactive manner.

B. Alshehry, W. Allen. Described a framework that consists of five layers. In first layer, the underdone network traffics was analyzed based on IP address of the sender. In second layer, the request speed was further analyzed to detect the DDoS attack. In third layer, an intelligent system was provided to assess the behaviors of the packets and detects the DDoS. As a preventive measures, in fourth layer load balancing was achieved whereas in the fifth layer as a proactive measure hopping port method was taken place.

T. Hirakawa, K.Ogura, B. B. Bista, & T. Takata proposed an evaluation of slow hypertext transfer protocol denial of service attacks in the cloud environments. The attacker kept his bull's eye on the hypertext transfer protocol servers to make it overloaded by slow frequent requests. However, most of the state-of-the-art works provides fewer effective results in detecting slow hypertext transfer protocol DoS attacks. The proposed method aims to reduce the vulnerability of these attacks by proposing the framework in which the sender with most request was pruned out of the network so called network saturation. The saturation point is fixed based on the pre-determined threshold values.

O. Yevsieieva, & S. M. Helalat group of authors proposed an assessment method for cloud environment to detect the slow hypertext transfer protocol denial of service and distributed denial of service attacks. Framework in which the sender with most request was pruned out of the network so called network saturation. The saturation point is fixed based on the pre-determined threshold values. Prior work focused only of denial-of-service attacks however attacks might attack in distributed way which causes severe effects. To cope up with this issue, the proposed work provides assessment method in which the sender request to the hypertext transfer protocol servers was analyzed and classified as normal or mischievous packets.

T. Hirakawa, K. Ogura, B. B. Bista, &, T. Takata proposed an efficient Resistance framework for cloud environment by detecting the slow HTPP denial of service attacks. Most of the state of the art work only assess the effect of these type of attacks however no work on preventing these attacks. The proposed work addressed these issues by proposing slow HTTP denial of service resistant framework in which sender request to the hypertext transfer protocol servers are analyzed in terms of duration time, and IP address. The senders whom are holding multiple requests in short period of time are marked as malicious and kicked out of the network.

M. Idhammad, K. Afdel, M. Belouch provides the detection method against slow HTTP distributed denial of service attacks using machine learning and entropy functions in cloud networks. The prior works faced with decreased true positive rates as the traffics in the network gone high. The proposed work solves this problem by sliding window algorithm based on time in which the sender packet feature was computed by entropy measure. The results are then forwarded to

random forest algorithm which classifies the features as normal or mischievous. The proposed work asses this work in CIDDS dataset which provides considerable results.

F. Ma, L. Zhang proposed a resilient management method for virtual machines in cloud environment using optimization algorithms The state of the art work limits with high energy consumption as it takes more time for migration. The proposed work alleviates those issues active mapping method in which the physical resources are optimally mapped to the corresponding virtual machines. In case of overload, the migration of VM to other VMs was done by considering the state of the VM and QoS of the user using threshold metho and sliding window algorithm.

A. Al-Dulaimy, W. Itani, R. Zantout, A. Zekri proposed the energy efficient virtual management method for cloud environment. The proposed method provides the framework which adapts the network dynamicity based on optimization algorithms and threshold methods. The proposed method considers parameters for VM management. The host which are under load are detected, host which are overload are detected. To reduce the frequent migration, the proposed work optimally chooses the VM to be migrated to overloaded/under loaded host.

Authors	Objectives	Methods/Algorithms	Disadvantages
O. Abdel et al	To improve the efficiency of DDoS attack detection based on load distribution	Trust based Maximin and Minimax model	• Faced with high complexity as the traffics gets increased
X. Chai, et al	To trade-off the security and cost constraints by integrating MTD and Q-learning	Deep Q-Learning algorithm	• Leads to high resource and energy consumption
K.A. Simpson et al	To improve the attack detection by performing per host-based reinforcement learning approach	Direct control reinforcement learning	• High time consumption as the assessment of every flow gets increases
K . J. Singh et al	To reduce the complexity during DDoS detection using fuzzy based genetic algorithm	Fuzzy logic and genetic algorithm	• Provides inconsistency in results
H.S. Mondal et al	To improve the attack detection accuracy by fuzzy logic	Fuzzy Inference System	• Inefficient method as the network in real time was dynamic in nature
M.S & P. Viswanathan et al	To reduce the energy consumption and resource wastage by optimal VM selection	Consumption factor based VM selection	• Considered only limited metrics for VM migration
S. Sambangi et al	To minimize the complexity during DDoS detection by machine learning methods	Multiple Linear Regression	• Limits with ineffective features selection as some of the important features are missing
S. Vetha et al	To maximize the DDoS detection strategy by performing game theory-based method in cloud environment	Trust based maximin model	• Provides inconsistent results when the traffics increases

A. Sahi et al	To maximize the cloud environment security against DDoS TCP flooding attacks	Two stage method	• Limits with poor detection accuracy
B. Alshehry et al	To improve the security measures in cloud environment by proactive DDoS detection method	Five-layer approach	• Faced with high computational complexity
T. Hirakawa et al	To reduce the effect of slow hypertext transfer protocol DoS attacks in cloud by network saturation	Threshold values	• Not provides effective results
O. Yevsieieva et al	To assess the impression of slow hypertext transfer protocol denial of service and distributed denial of service attacks in cloud environment	Analysis of sender's requests	• Not sufficient parameters for detection leads to less detection accuracy
K. Ogura et al	To maximize the security against slow HTTP denial of service attacks by resistance framework	Analyse sender request	• Only considers two features which provides inconsistent results
M. Idhammad et al	To increase the true positive rate against slow HTTP distributed denial of service attacks by machine learning methods	Entropy measure and Random Forest ensemble algorithm	• Provides fewer effective results in real time environments
F. Ma et al	To improve the QoS by optimally managing the VMs in cloud environment	Sliding window algorithm and threshold method	• Not enough metrics for optimal migration
A. Al-Dulaimy et al	To reduce the energy consumption by optimal VM management method	Optimization algorithms	• High possible to traps with local minima

TABLE 1: Related Works.

4. PROPOSED WORK

In this section, the proposed work and existing maximin work in detail. To be more distinctive, both the works are explained in detail with its suitable diagram and equations respectively. This is split up into two sub-sections such as,

- Proposed RL-DDoS
- Existing Maximin Model

The proposed work mainly focused on providing security to the cloud environment against DDoS attacks using RL.The adoption of RL for DDoS detection ensures adaptability based on the network dynamicity and also provides the error free detection results in the real time environment. The key entities in the proposed work define users which include normal and malicious users, an authentication server, IDS system, physical servers, and virtual machines. To be more distinctive, the proposed work provides this section as three sub-sections which are described below. Fig 3 represents the overall proposed architecture with entities in detail.

- Multi Factor Authentication
- DDoS Attack Detection
- Risk Aware VM Isolation

4.1 Multi Factor Authentication

The users in the cloud environment are authenticated initially to prevent the DDoS attacks. As the network traffic gets increased, without checking the user legitimacy to the network leads to high malicious traffic which also causes scalability issues. The proposed work adopts SHA-512 algorithm for authentication which ensures the integrity of the user information by its collision resistant nature. Before authentication, the user needs to their information such as user ID, Password (Pwd) which are provided as steps given below,

- Step 1: The cloud users (C_u) are registered with their credentials such as user ID, password to the authentication server (∀),

$$C_u(ID, Pwd) \rightarrow \forall \quad (3)$$

- Step 2: The ∀ store the corresponding credentials in hashed manner using SHA-512 algorithm. The SHA-512 adopts two sub-steps to converts the credentials into hash values which are provided below,

    1. The first step is to add the chunks of bits to the input credential information, so that the input credential is corresponding with 890 mod 1024.
    2. The second step is to add the redemption value to input credentials to recover the input as hash value. This can be formulated as,

$$\forall \rightarrow \forall \left[R\left[\hat{\mathrm{H}}(ID, Pwd)\right]\right] \quad (4)$$

- Step 3: Once the input credentials are hashed, the ∀ is acknowledged with providing unique random number (RAN) through secure channel which indicates the successful registration which can be formulated as,

$$\forall(RAN) \rightarrow C_u \quad (5)$$

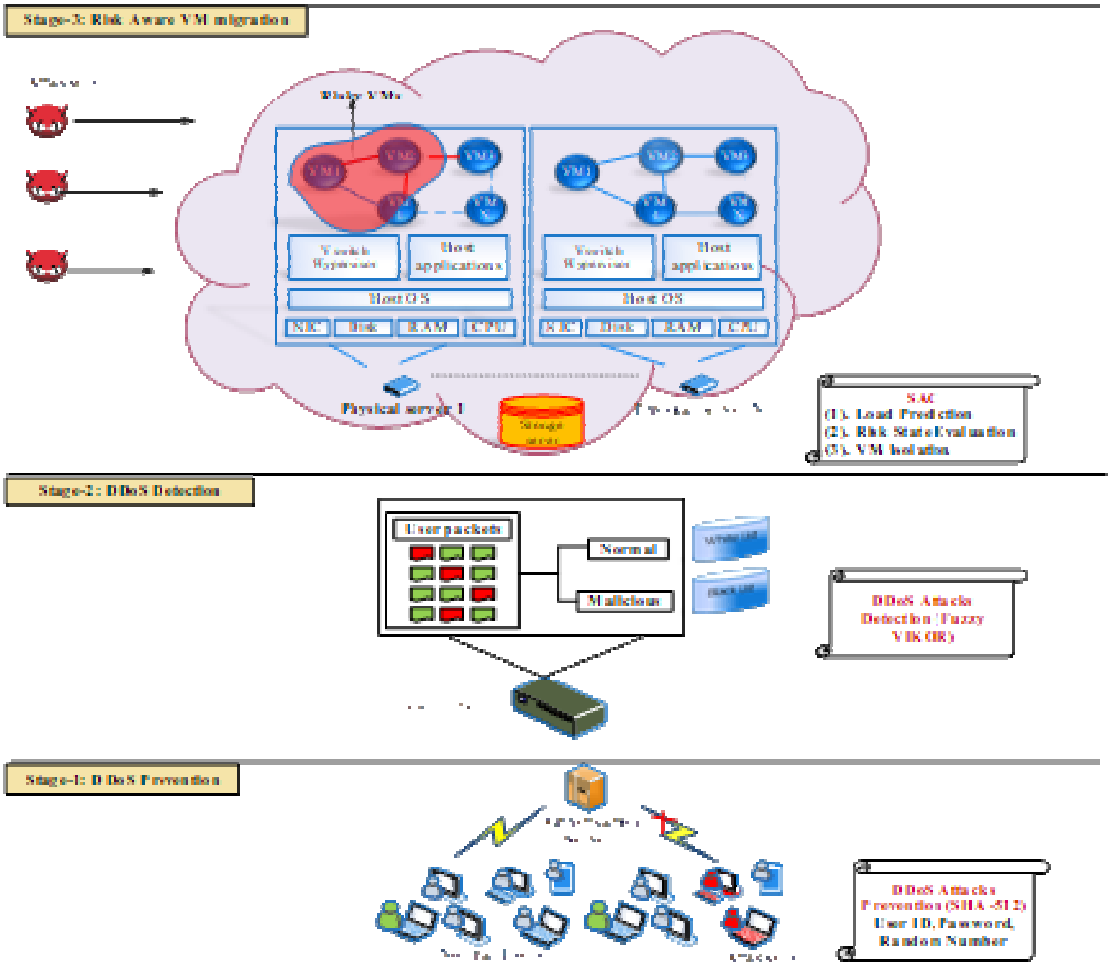

FIGURE 3: Proposed RL-DDoS Cloud Architecture.

The registered users are only allowed for authentication. Once completed registration, the users are authenticated through their multi factors to the authentication server using SHA-512. The steps involved in authentication are provided below,

- Step 1: During authentication, the users are verified with multiple factors by the ∀ which is represented as below,

$$C_u(ID, Pwd, RAN) \rightarrow \forall \quad (6)$$

- Step 2: The users submit their factors in hashed manner to the ∀ using the SHA-512 which is denoted as,

$$\hat{\mathrm{H}}(ID, Pwd, RAN) \rightarrow \forall \quad (7)$$

- Step 3: Upon receiving the hashed values, the ∀ compares each factor with already registered factors to ensure the user authenticity which is formulated below,

$$\forall = \begin{cases} \hat{\mathrm{H}}(ID) = R\left[\hat{\mathrm{H}}(ID)\right], \text{ "Y" or "N"} \\ \hat{\mathrm{H}}(Pwd) = R\left[\hat{\mathrm{H}}(Pwd)\right], \text{ "Y" or "N"} \\ \hat{\mathrm{H}}(RAN) = R\left[\hat{\mathrm{H}}(RAN)\right], \text{ "Y" or "N"} \end{cases} \quad (8)$$

From the above equation, $\hat{H}(ID)$ denotes the hashed credentials during authentication whereas, $R[\hat{H}(ID)]$ indicates the hashed credentials during registration, "Y" or "N" represents the weather the hashed values are matched or not. If both are matched, then "Y" otherwise "N". On the other hand, if any one of the credentials are not matched, the access for the corresponding user will be denied. Fig 4 represents the proposed Multi factorin detail.

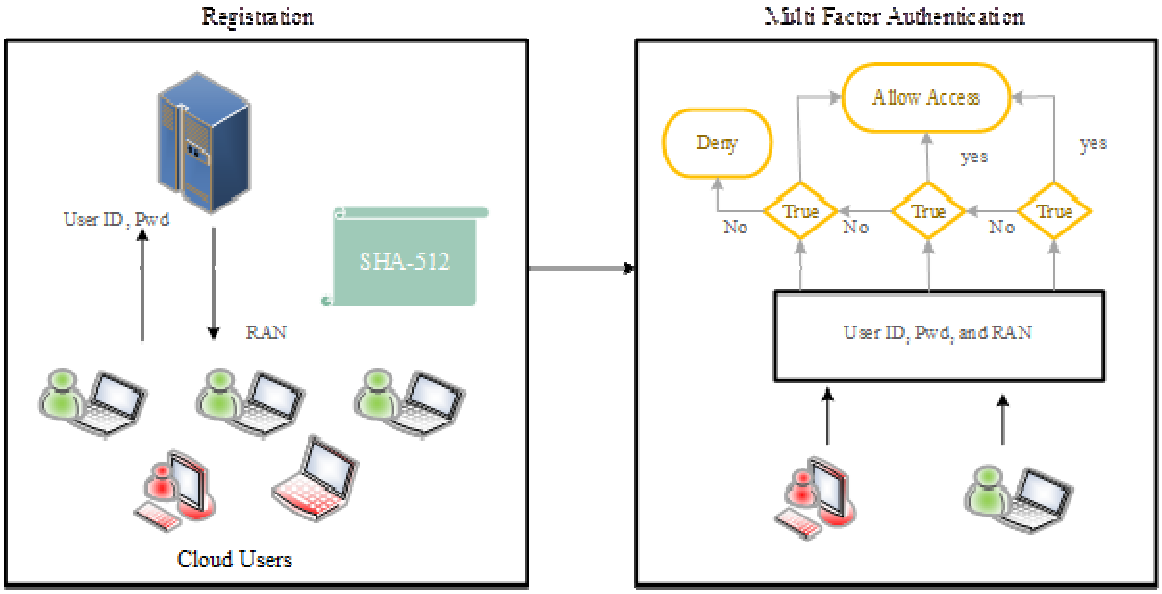

FIGURE 4: Registration and Multi Factor Authentication.

4.2 DDOS Attack Detection

The above step provides the preventive measure of DDoS attacks from outside. However, the DDoS attacker might compromise the users and performs DDoS through that compromises devices. Hence it needed to be detected in order to safeguard the network. For DDoS detection, the proposed work adopts Fuzzy VIKOR algorithm which analyze the user packets and gets the output as normal packets or mischievous packets. The proposed work further divides this DDoS detection into two sub-sections such as feature selection, and data packet classification.

- **Feature Selection:** Before analyzing the user data packets, the optimal user data packets must be selected to improve the detection accuracy and at the same time it also reduces the complexity and redundancy during classification. With lot number of user data packet features, we optimally select the best features which resides the lot of information in it. The optimal features selected by the proposed work for feature classification are Source Destination IP Address, IP Address, Destination Port, Source Port, Protocol Type and Packet Size. The selected features are provided to data packet classification stage.

- **Data Packets classification:** The selected features are forwarded to the Fuzzy VIKOR. The Fuzzy VIKOR is the multi-criteria decision-making algorithm which makes the best decision among the multiple criteria. The fuzzy logic is used to improve the results precision as in some case there might be indefinite numerical quantities are analyzed. The procedure of Fuzzy VIKOR is provides as phases in the proposed work as,

Phase 1: In this phase, based on the elected features the decision matrix was constructed (D) which also holds the bipolar fuzzy number and alternatives. Further, by adopting cross entropy

function the weight value is provided to each feature. The steps involved in this phase are provided below,

$$D = \begin{matrix} atn_1 \\ atn_2 \\ atn_{l-1} \end{matrix} \begin{bmatrix} e_{11} & e_{12} & \dots e_{1q} \\ e_{21} & e_{22} & \dots e_{2q} \\ e_{(l-1)1} & e_{(l-1)2} & \dots e_{(l-1)q} \end{bmatrix} \quad (10)$$

From the above equation, atn denotes the substitutes which ranges from $(1, \dots, \mathrm{l}-1)$, and e denotes the different criteria (i.e., features). The function to represent the bipolar fuzzy numbers can be formulated as,

$$e_{\gamma\delta} = \langle [x_{\gamma\delta 1}, x_{\gamma\delta 2}, ..] | [z_{\gamma\delta 1}, z_{\gamma\delta 2}, ..] \rangle \quad (11)$$

Where, x, y denotes the bipolar fuzzy numbers. The ranking of bipolar fuzzy number can be formulated as,

$$F_{\gamma\delta} = \left(\left[\frac{x_{\gamma\delta 1} + x_{\gamma\delta 2} + ..}{4} \right] + \left[\frac{-x_{\gamma\delta 1} - x_{\gamma\delta 2} + ..}{2} \right] \right) - \left(\left[\frac{z_{\gamma\delta 1} + z_{\gamma\delta 2} + ..}{4} \right] + \left[\frac{-z_{\gamma\delta 1} - z_{\gamma\delta 2} + ..}{2} \right] \right) \quad (12)$$

From the above fuzzy rank ($\mathrm{F}_{\gamma\delta}$) equation, using cross entropy function for every feature the weight values are assigned as prognosis value $(\aleph_{\gamma\delta})$ which can be formulated as,

$$\aleph_{\gamma\delta} = \frac{F_{\gamma\delta}}{\sum_{\gamma=1}^{(l-1)} F_{\gamma\delta}} \quad (13)$$

For the above equation, the cross-entropy function can be formulated as,

$$CEF_{\delta} = -\sum_{\gamma=1}^{l-1} \frac{1}{P} \log_2 \aleph_{\gamma\delta} \quad (14)$$

Based on the above equation, the divergence degree was calculated which can be represented as,

$$\Phi_{\delta} = 1 - \xi_{\gamma\delta}, \dots \dots \dots \delta = 1,2,3 \quad (15)$$

From the above equation, the weights of each feature can be computed by,

$$We_{\delta} = \frac{\Phi_{\delta}}{\sum_{\delta=1}^{3} \Phi_{\delta}} \quad (16)$$

Phase 2: In this phase, based on the weight values the ranking of best and worst features are computed, and best solution is provided. The equations involved in this phase are provided below,

$$\mathrm{j}_{\delta}^{*} = \max_{\gamma}\{\mathrm{j}_{\gamma\delta}\}, \mathrm{j}_{\delta}^{-} = \min_{\gamma}\{\mathrm{j}_{\gamma\delta}\} \quad (17)$$

$$\mathrm{j}_{\delta}^{*} = \max_{\gamma}\{\mathrm{j}_{\gamma\delta}\}, \mathrm{j}_{\delta}^{-} = \min_{\gamma}\{\mathrm{j}_{\gamma\delta}\} \quad (18)$$

From the above equations, $j_\delta^* \; and \; j_\delta^-$ denotes the best and worst selection respectively. From that feature, the increased group utility function (GU_γ) and least regret of the specific opponent (OP_γ). The values of GU_γ and OP_γ can be formulated as,

$$GU_\gamma = \sum_{j=1}^{3} We_\delta \frac{j_\delta^* - j_{\gamma\delta}}{j_\delta^* - j_\delta^-} \tag{19}$$

$$OP_\gamma = \max_\delta \left(We_\delta \frac{j_\delta^* - j_{\gamma\delta}}{j_\delta^* - j_\delta^-} \right) \tag{20}$$

From the above equations (19) and (20), O_γ can be formulated as,

$$O_\gamma = x \frac{GU_\gamma - GU^*}{GU^- - GU^*} + (1 - x) \frac{OP_\gamma - OP^*}{OP^- - OP^*} \tag{21}$$

Where, $GU^* = \min_\gamma \{GU_\gamma\}$, $GU^- = \max_\gamma \{GU_\gamma\}$, $OP^* = \min_\gamma \{OP_\gamma\}$, and $OP^- = \max_\gamma \{OP_\gamma\}$. By mathematically solve the above values, the proposed work obtains the final output equation which can be formulated as,

$$O\left(I^{(2)}\right) - O\left(I^{(2)}\right) \geq \mathfrak{D}O \tag{22}$$

Where, $I^{(2)}$ is the 2nd position alternative. By solving the above equation, we can obtain the normal and malicious user packets. The pseudocode of the proposed Fuzzy VIKOR is provided below,

```
Pseudocode
Fuzzy VIKOR based DDoS Detection ()
Start
    Phase 1:
      Construct Decision matrix D using (10)
      Represent bipolar fuzzy numbers using (11)
      Rank the fuzzy bipolar value using (12)
      Assign weight for each features using (16)
End
    Phase 2:
      Compute best and worst feature ratings using (17)-(18)
      Compute the group utility function using (19)-(20)
      Compute fuzzy VIKOR index using (21)
      Rank the features based on (GU_γ) and O_γ to get final output
  End
End
```

Pseudo code: Fuzzy Vikor.

From the above pseudo code, the fuzzy VIKOR is utilized for DDoS detection in which two phases are employed. In first phase, based on the adopted features for DDoS detection the fuzzy decision matrix is constructed. From that decision matrix, the bipolar fuzzy numbers are determined to obtain the best alternatives, and also ranking of fuzzy bipolar number is taken to further enhance the results. The ranked features are assigned with corresponding weights. The assigned weights are forwarded to second phase from which the best and worst features in the weights are rated. With that feature ratings, a group utility is computed which acquires and interprets on feature importance in decision making. The group utility is utilized to compute fuzzy VIKOR index. From that index, the fuzzy variables are arranged in descending order to obtain the final result.

The adoption of multi criteria decision making (i.e., fuzzy VIKOR) algorithm for proposed DDoS detection not relies on single input rather it grabs the importance of other features are provides effective and accurate outputs. It also provides efficient results in complex and dynamic scenarios which is well suited in the case of cloud environment. In contrast, the existing maximin model adopts maximin/minimax game theory model for DDoS detection in which the defenders try to maximize their strategy to minimize the attacker attacking probability based on trust scores on resource usage which affects the efficacy for the DDoS detection in the dynamic real time cloud environment. The adoption of maximin game theory model rather than the proposed fuzzy VIKOR based model would not provide the worst-case scenario results which makes the player to trap on worst case scenario and leads to ineffective results. In addition, the features utilized by the existing work such as CPU utilization rate, network bandwidth consumption, and memory consumption rate for DDoS detection is less when compared to the proposed work features.

4.3 Risk Aware VM Isolation

For the legitimate data packet from the above detection method, the proposed work allocated resources dynamically through VMs. By predicting the VMs state such as,

- **CPU capacity:** It is defined as the amount of data that the VM can process in single cycle in terms of bits/cycle. To be more specific, the speed of computation in the hardware device of the system.
- **Memory storage:** The capacity of the VM to store the user is known as memory storage. The VM storage can be mapped to the physical machine in which the VM storage can be modifiable regardless of underlying hardware.
- **VM states:** The VM sates are classified into three types such as, idle state, busy state, and running state. In idle state, the VM is not detached rather it runs and consumes resources which can be used for provide additional resource to the other physical machines. In busy state, the VMs are hosting any user application. In running state, similar to the busy state VMs resource is completely filled.

Based on the above parameters, the resource allocation is performed. Eventually, the VMs are dispersed optimally using metrics such as size of VM, type of VM, size of host, and type of host. On the other hand, cyber crooks try to manipulate the VMs from the outside which leads to resource unavailability and resource wastage. To cope up with these issues, we perform probability based VM isolation in terms of risk and no risk.

- VM Isolation in Probability of Risk

In case of risk scenario (i.e., VM is at overload stage or malicious cyber crooks tries to manipulate the VM) we perform VM migration and isolation respectively. The VM states such in terms of idle state, busy state, running state, and security state are continuously monitored to determine the probability of risk. The risk probability can be formulated as,

$$Pr_{OL}^{ris}(1-\alpha_{Res}) \tag{23}$$

From the above equation, Pr_{OL}^{ris} denotes the overloaded and manipulation risk of the VM respectively, while α_{Res} illustrates the resources that are previously processed in the VM. The status of the VM can be computed by,

$$\sigma_{Res}^{*}(y|\vartheta_{Res},\alpha_{Res}) = \frac{1}{\alpha_{Res}\sqrt{2\pi}}\int_{-\infty}^{y} e^{-\frac{(y-\vartheta_{Res})^2}{2\alpha_{Res}^2}}\,dy \tag{24}$$

From the above equation, $\vartheta_{Res}, \alpha_{Res}$are the probability function standard deviations respectively, σ_{Res}^{*} the any time risk probability of VM. Based on the above, one can define the VM probability at risk as $Pr_{OL}^{ris} > 0.5$. This determination shows that, when the probability surpasses 0.5 then the VM is at risk.

Even though we predict the VM states and security constraints some of the high potential cyber a crook manipulate the VMs in the cloud environment and makes them overload and disseminate to other legitimate VMs. To avoid that issue, we perform isolation of compromised VMs based on the risk probability. For that we adopt Soft Actor Critic (SAC) algorithm which learns the environment state and perform actions according to that. The SAC is the off-policy algorithm which is highly suitable for continuous events. The SAC tries to maximize the policy by capitalizing the entropy function without changing the objective function. The entities involved in SAC are actor and critic. The actor observes the state (St) of the environment and performs corresponding action based on the state space whereas the critic observes the St and tries to maximize the total expected reward function. The SAC also performs regularization of entropy which enhances the convergence and learning rate which can be formulated as,

$$\pi^* = \arg\max_{\pi} \sum_t E_{(St_t, Ac_t)} \sim p_\pi \left[Re(S_t, Ac_t) + \Theta H(\pi. | St_t) \right] \quad (25)$$

Where, S_t, Ac_t denotes the state and action, Re denotes the reward for state and action, $H(\pi. | St_t)$ indicates the policy of the entropy, and Θ gives the trade-off between exploitation and exploration states respectively. The state value function, Q-value of the soft function, and policy which is traceable respectively that can be represented as $V_\psi(st_t)$, $Q_\theta(St_t, Ac_t)$, and $\pi_\phi(Ac_t | St_t)$.The residual error can be decreased by train the state value function which is computed below,

$$K_v(\psi) = E_{St_t \sim RBF} \left[\frac{1}{2} \Big(V_\psi(St_t) - E_{\mathrm{Ac}_t \sim \pi_\phi} \big[Q_\theta(St_t, Ac_t) - \log \pi_\phi(Ac_t | St_t) \big] \Big)^2 \right] \quad (26)$$

Where, RBF denotes the replay buffer which provides the previously stored state and actions in the environment. The valuation of gradient can be formulated as,

$$\widehat{\nabla_\psi K_v(\psi)} = \nabla_\psi V_\psi(St_t) \big(V_\psi(St) - Q_\theta(St_t, Ac_t) + \log \pi_\phi(Ac_t | St_t) \big) \quad (27)$$

By adopting Stochastic Gradient Function (SGF), the Q value function can be optimized which can be formulated as,

$$\widehat{\nabla}_\theta K_Q(\theta) = \nabla_\theta \mathrm{Q}_\theta(Ac_t, St_t) (Q_\theta(St, Ac_t) - Re(St_t, Ac_t) - \blacksquare V_\psi(St_{t+1}) \quad (28)$$

With the above equations, by learning the policy of the parameter the finest policy van be achieved which can be formulated as,

$$K_\pi(\phi) = E_{St_t \sim RBF, \epsilon_t \sim N} \left[\log \pi_\phi \big(f_\phi(\epsilon_t; St_t) \big| St_t \big) - Q_\theta \left(St_t, f_\phi(\epsilon_t; St_t) \right) \right] \quad (29)$$

The trained policy are stored in the actor to provides the error free results by consuming minimized power and execution time respectively. The Fig 5 and pseudocode 1 represent the risk aware VM isolation in the cloud environment.

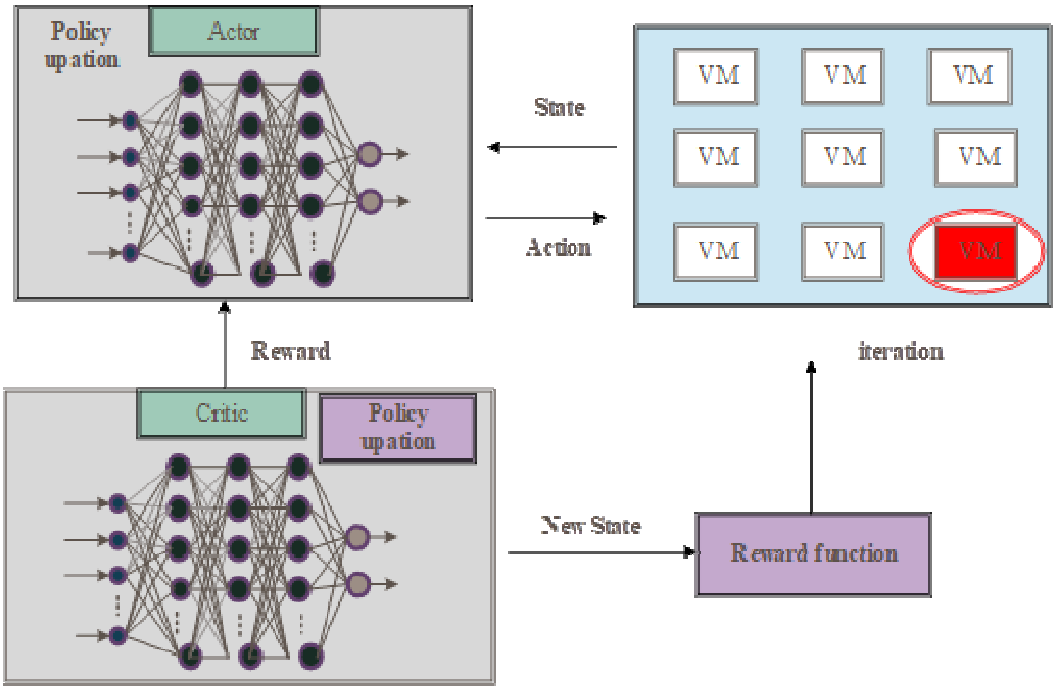

FIGURE 5: SAC based VM isolation.

Pseudocode
SAC based Risk Aware VM Migration ()
Set Local Networks
Set Target Networks
Set $\propto\rightarrow RBF$
For Every Iteration **do**
For Every environment **do**
$Ac_t \sim \pi_\phi(Ac_t \mid St_t)$
$St_{t+1} \sim p_\pi(St_{t+1} \mid St_t, Ac_t)$
$RBF \leftarrow RBF \cup \{(St_t, Ac_t, Re(St_t, Ac_t), St_{t+1})\}$
For Every Gradient Step **do**
$\theta_i \leftarrow \theta_i - \partial_\theta \nabla_\theta, M(\theta_i)$
$\phi \leftarrow \phi - \partial_\phi \nabla_\phi, M(\phi_i)$
$\psi \leftarrow \psi - \partial \nabla_\psi M(\psi)$
End
End
End

PseudocodeSAC based VM Isolation.

Further, the proposed work tries to reduce the SLA ($\tilde{U}_t$) violation for enhancing the QoS which can be done by,

$$\tilde{U}_t = \sum_{i=1}^{n} (€_{ri} - €_{ai}) \quad (30)$$

Where, $€_{ri}$ denotes the MIPS which is requested, $€_{ai}$ is the MIPS which is allocated. For the users or CSPs who can violate the SLA the forfeit function can be formulated as,

$$P_t(\bar{\mathrm{U}}_t) = \frac{\bar{\mathrm{U}}_{t_{t+1}}}{\bar{\mathrm{U}}_{t_t}} \quad (31)$$

- VM Isolation in Probability of No Risk

For the no risk scenario (here risk means when the VM gets overloaded or it is vulnerable to DDoS attacks by the external cyber crooks) there is no need for isolate the VM. The probability of no risk can be determined when the probability of risk surpasses below 0.5 $Pr_{OL}^{ris} < 0.5$.

5. MAXIMIN MODEL

In this method, the deprivation of cloud resources by the DDoS attacker was diminished by adopting trust-based game theory model. To be more specific, this work focused on distribution of finest load among VMs in the environment with consideration of DDoS attackers. This work composed of two sequential processes such as,

- Trust based VM Relationship Construction
- Hypervisor based Maximin Model

5.1 Trust based VM Relationship Construction

The trust between cloud entities ensures security against DDoS attacks in the cloud environment. Here, the hypervisor is responsible for enabling trust relationship among VMs. For that, two types are trust was calculated as direct trust, and direct trust which are explained below,

Direct Trust: This type of trust was evaluated by the hypervisor in which the behavior of the targeted VM is continuously monitored. The targeted VM was considered as a compromised when it possesses high resource usage. Here, the VM bandwidth consumption, utilization of CPU rate, and consumption of memory rate were considered as resources. This work adopts interquartile range measure to perceive the unwanted usage of the VM's resource. The monitored data was chunked as quartiles by the interquartile measure which was denoted as $iQ_1, iQ_2, and\ iQ_3$. The iQ_1 denotes the given data value was twenty-five percentage lesser than it, iQ_2 represents the median value of the given set of data, iQ_3 and illustrates the given data value was twenty-five percentage higher than it. By differencing the iQ_1 from the iQ_3 interquartile range was obtained. The main reason for adopting interquartile measure was, it possesses low computational complexity due to its lightweight nature, and highly precision to data outliers.

- By monitoring the VM's resource usage, at time period ti the hypervisor calculates the median consumption of the resource usage which is denoted as,

$$Med_{VM}(t) \leftarrow ReU_{VM} \quad (32)$$

Where, ReU_{VM} resources of the VM such as VM bandwidth consumption, utilization of CPU rate, and consumption of memory rate, and Med_{VM} median range utilization of VM resource.

- After finding the $Med_{VM}(t)$ the 1st and 3rd quartile range of the resource usage was determined which can be denoted as,

$$Med_{VM}(t) = \begin{cases} iQ_1 > 25\% \\ iQ_3 < 25\% \end{cases} \quad (33)$$

- By computing the median value of 1st and 3rd quartiles, the interquartile range was computed which can be formulated as,

$$iQr_{VM}(t) = \left(iQ_{3_{VM}}(t) - iQ_{1_{VM}}(t) \times 1.5\right) \quad (34)$$

From the above equation, 1.5 denotes the any resource that consumes higher than 1 and half of the iQ_3 is noted as an infrequent consumption which can be denoted as,

$$UL_{VM}(t) = iQr_{VM}(t) + iQ_{3_{VM}}(t) \quad (35)$$

Where, $UL_{VM}(t)$ denotes the upper limits of VM infrequent consumption which can be determined by summing up the $iQr_{VM}(t)$ with $iQ_{3_{VM}}(t)$. The eqn () is considered as the normal utilization range. In case if any of the VM consumes beyond that limit $(i.e. UL_{VM}(t) < VM)$ the VM's trust value will be less and marked a overconsumed VM. On the other hand, if no VM consumed beyond the determined limit then the belief for that corresponding VM is set to one.

Indirect Trust: Other than the direct trust, to establish more trusted relationship the feedback about the monitored VM was also acquired by the hypervisor. These feedbacks are collected from the neighboring VMs or neighboring hypervisor whom already hosted that VM. The feedback collection not only improves the security level but also enhance the QoS and SLA requirements in the cloudenvironment. On the other hand, the hypervisor collects the behavior of monitored VM not only related to the DDoS but also collects feedback about other related attacks. If the collected feedbacks were not satisfied then the corresponding VM is marked as compromised else ready for hosting.

Bayesian based Trust Accumulation: After acquiring both the direct and indirect trust, the hypervisor adopts Bayesian interference to accumulate the trust. The utilization of Bayesian interference solves the vagueness problems by probability-based distributions. The Bayesian based direct and trust was determined by,

$$_{hyp}^{VM}Beleif = \sum_{i=1}^{m} \frac{fb_{so}^{VM} + n\text{œ}}{2n} \quad (36)$$

Where, œ denotes the direct belief by the hypervisor (hyp) on the VM, Rec_{so}^{VM} represents the indirect belief score by feedbacks acquired from the various sources (so) about the VM. For successful feedbacks, the VM or hypervisor receives rewards.

Trust for Newly Installed VM's: Trusting newly installed VM was a major security concern as there is no past information and resource usage of the VM. This makes difficult for the hypervisor to construct relationship with that. This work considered this issue as a bootstrapping issue. The issue was solved by the advocacy method which used decision tree classifiers to solve that issue. Any VM which was newly deployed, the hypervisor which was to be hosted that newly installed VM would set the initial trust scores and seeks help to the other hypervisor or VM. The hypervisor sent request to them to endorse the newly installed VM. Any willingness hypervisor or VM whom had enough resources to endorse the new VM's would train the decision tree classifier in terms of initial trust score, installer name, and location. By collecting the advocacy results from the different hypervisors or VM's, their results are aggregated using Bayesian interference as in eqn (35). If the results are successful, then the newly VM was hosted by the hypervisor else revocated.

5.2 Hypervisor based Maximin Model

Upon computing the trust scores for the VM's, the maximin model was adopted to mitigate the DDoS attacks. For that, efficacy function (EF) is computed. The EF is computed by,

$$EF^t(hyp) = \sum_{VM} \frac{z(VM) \times \propto_{[t_1,t_2]}}{_{hyp}^{VM}Beleif} \quad (37)$$

where, $z(VM)$ denotes the VM's worth in terms of its application sensitivity and cost, $\propto_{[t_1,t_2]}$ intrusion detection agent whom running inside the hypervisor at window of time $t_1 \; and \; t_2$ respectively. After determining the efficacy function, the maximin game theory was modelled as,

Attacker-Hypervisor minimax/maximin Game: The tuple of attacker-hypervisor game can be modelled with set of players and their objective function that can be denoted as $[Att, Hyp, VM_{lp}, Att_{VM}, EF]$. From the above set, Att denotes the attacker, Hyp is the hyper visor, VM_{lp} load distribution of VM which is noted as mixed game procedure of the VM, Att_{VM} is the probability of attacker tries to host the VM which is noted as mixed game procedure attacker, and EF denotes the hypervisor utility function.

The attacker objective is to minimize the hypervisor detection rate against the DDoS attacks and the hypervisor reward which is represented as,

$$\text{argmin}_{Att_{VM}} EF(Hyp) \tag{38}$$

Eventually, the hypervisor tries to maximize the detection rate by minimizing the attacker objective which can be represented as,

$$\text{argmax}_{VM_{lp}} \min_{Att_{VM}} EF(Hyp) \tag{39}$$

By maximizing the detection rate, the attacker objective function would be reduced results in effective DDoS detection. The comparison of proposed and existing works in terms of processes, and algorithms are provided in the form of table II below.

Processes	Proposed RL-DDoS cloud	Maximin Model
DDoS prevention	The proposed work prevents DDoS by authenticating the users with their multiple factors such as ID, password, and random using SHA-512 algorithm.	Here, only trust scores were employed in which the trust scores pre-determinedly set for the newly deployed VMs and computes their degree of trust using decision tree-based endorsement method.
DDoS detection	The proposed work detects the DDoS attacks by monitoring and acquiring the user related metrics such as parameters IP Address, Destination IP Address, Source Port, Destination Port, Packet Size and Protocol Type. The features are assessed by the fuzzy VIKOR decision making algorithm for detecting the abnormal user behaviors.	In this work, VMs are continuously monitored directly and indirectly. The direct monitoring also relies on trust in which the CPU utilization rate, network bandwidth consumption, and memory rate was considered. In indirect monitoring the neighbors feedback was acquired. Both are acquired using Bayesian model. Further, the maximin game model was played between hypervisor and attacker to detect the DDoS attacks.
DDoS Mitigation	The proposed work mitigates the detected by adopting VM migration and isolation techniques. Here, for DDoS mitigation SAC algorithm is utilized which optimally finds the compromised VM and isolates them, and also migrates the overloaded VM to optimal VM based on the already presented VM states.	Here, VM loads are optimally distributed based on the CPU utilization, network bandwidth consumption, and memory consumption rate. However, this process would be coming under mitigation model.

TABLE 2: Proposed RL-DDoS Vs Existing Maximin Model.

6 EXPERIMENT RESULTS

This section explains about the proposed experimental results in terms of simulation setup, comparative analysis, and research summary. The detailed explanation of the three sub-sections are provided below.

6.1 Simulation Setup

The proposed work is tested and simulated using Cloudsim simulation tool with several simulation entities such as physical node, virtual machine, data center, and cloudlet. The adoption of Cloudsim for simulation provides the resilient simulation environment in terms of user specification and run time environment. The proposed work simulated this research work using the system settings of hardware configuration and software configuration. For hardware configuration, CPU used is Intel ® Core ™ i5-4590s @ 3.00 GHz, hard disk used is 500GB, and RAM used is 8GB. For software configuration, the Language used is JAVA, OS used is windows 10 pro, IDE used is NetBeans 12.3, and simulator used is Cloudsim. Table III gives the proposed work simulation parameters and their specifications.

Simulation Entity	Description	Value
Virtual Machine	# of Processing unit	4
	MIPS	500-2000
	RAM	128-2048
	Bandwidth	500-1000
	Storage	11 TB
Data center	# of data center	10
	# of hosts	2-6
Physical Node	MIPS	2500-3000
	Memory	32 GB
	Storage	1 TB
	Bandwidth	10 GB/s
Cloudlet	Size of the file	500
	# of task	50-500
	Average Length	50,000

TABLE 3: Simulation Parameters.

6.2 Comparative Analysis

The proposed RL-DDoS Cloud is compared with existing work named Maximin [21] in terms of evaluation metrics such as response time, latency, throughput, and resource utilization. These metrics are used to improve the proposed work performance than the existing work. The detailed description and their graphical results are provided below,

1. Response Time Comparison

The amount of time taken for processing all the data packets in known as response time (rs). Generally, a good model must have less response time. The response time can be formulated as,

$$rs = \sum_{n=0}^{N=0} T_{Dpack} \quad (32)$$

Where, T denotes the time taken for complete the process of given data packets $Dpack$, and N denotes the number of nodes in the network. Fig 6 represents the response time comparison of proposed and existing maximin work respectively. From the figure, it is shown that, when the number of tasks increases response time also increases. Among that our proposed work achieves less response time than the existing works this is due to effective handling and detection of data packets using fuzzy VIKOR algorithm. Due to adoption of this process, the user packets with requests are handled quickly as possible using packet related features such as Source Destination IP Address, IP Address, Destination Port, Source Port, Protocol Type and Packet Size classify the packets as normal and malicious. Whereas, the existing maximin work limits with intelligent behavior as it considers only trust based on CPU utilization, network

bandwidth usage, and memory rate utilization rather than selecting real time features. However, the existing maximin work concentrates only on the VM's usage not focusing on user side constraints which limits the response time of real time DDoS detection. From the graphical results, it is shown that when the number of tasks gone to 20 the proposed work achieves less response time of 160 ms whereas the existing maximin approach achieves high response time of 220 ms for the same number of tasks. With those results, it is clear that the proposed work achieves 40ms lesser than the existing maximin work.

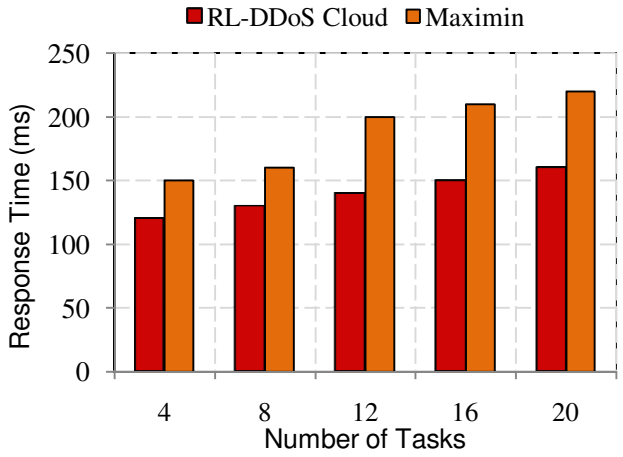

FIGURE 6: Number of Tasks Vs Response Time.

2. Latency Comparison

The amount of time taken for the data packet to reach its destination is known as latency. Typically, a good network must be latent free. The latency (Lat) can be formulated as,

$$Lat = \sum_{n=0}^{N} \frac{T_{sou}}{T_{des}} \quad (33)$$

Where, T_{sou} denotes the amount of time the data packet sent by the source while T_{des} is the amount of time the data packet received by the receiver. Fig 7 represents the comparison of number of tasks vs latency in which the when the number of tasks increases the latency also increases. However, the proposed work achieves less latency than the existing maximin work. The reason for achieving the low latent nature of the proposed work is, as its proposes multi factor authentication for the user initially. The adoption of multi factor authentication method reducing the processing latency during further process (i.e., DDoS detection). Whereas, the existing maximin approach considers hypervisors as legitimate whom hosting the VMs based on the trust. However, without considering the hypervisor legitimacy increasing the processing latency of the cloud service providers as the illegitimate hypervisor possess unwanted tasks to the legitimate hypervisor leading to high processing latency in the cloud environment. From the numerical results, it is shown that when the number of tasks increases to 20 the proposed work achieves the less latency of 80 ms while the existing work maximin achieves high latency of 120 ms for same number of tasks.

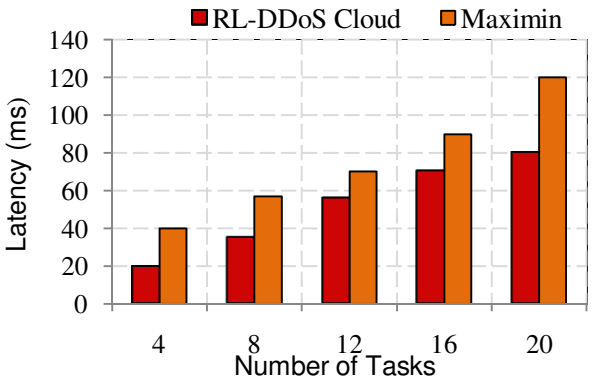

FIGURE 7: Number of Tasks Vs Latency.

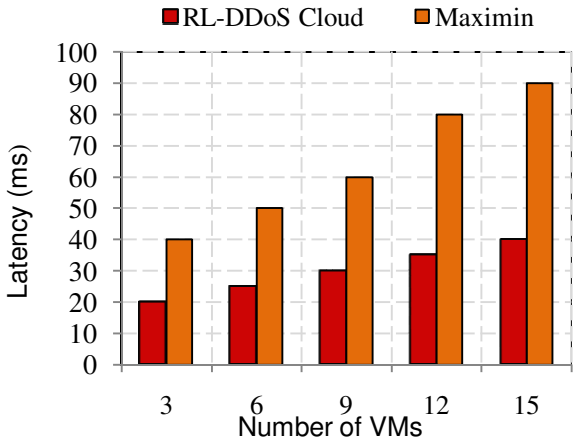

FIGURE 8: Number of VMs Vs Latency.

Similarly for the fig 8, when the number of VMs increasing, the latency also increasing. Among that our proposed work achieves less latency this is due to optimal VM migration and isolation using SAC. The proposed method selects the optimal VMs based on their states (i.e., idle, busy, running), VM size, host size, CPU utilization rate which reduces the processing latency of user request. Further, the compromised VMs are isolated which reduces the latency due to security breaches. The existing maximin model limits with only VMs CPU utilization rate, memory, and bandwidth consumption rate however the other effective metric are not considered leads to high latency in DDoS detection. The graphical results show that, the proposed work achieves less latency of 40 ms even though the VMs number are gone to 15 whereas the existing works achieves high latency of 90 ms for same number of VMs.

3. Throughput Comparison

Throughput (∀) is defined as the amount of user packets that delivered successfully to the destination which is calculated as bits per second. The formulation of through can be derived as,

$$\forall = \frac{Del^{pack}}{tot_{pack}} \tag{34}$$

where, tot_{pack} denotes the total number of packets, and Del^{pack} denotes the number of packets successfully delivered. The fig 9 represents the throughput comparison of proposed and existing work maximin respectively. From that figure, it is depicted that when the number of tasks increases throughput also increases this is due to proposed authentication and risk aware VM

migration and isolation respectively. For authentication, the proposed work adopts SHA-512 algorithm which secures user credentials thereby improving the throughput rate due to security breaches, and risk aware VM isolation and migration selects the optimal VM for secure migration and also isolates the VM that are compromised using SAC increases the packet success rate thereby improving the throughput rate. Whereas, the existing maximin approach lacks with considering user legitimacy rather is computes trust for the all the VMs and compute endorsement value to the newly allocated VMs by its location and provider only. However, only by the trust and endorsement the legitimacy was not effectively ensured results in high illegitimate traffic thereby leading to less throughput rate.The numerical analysis shows that, the proposed work achieves high throughput of 80 Kbps when the tasks increase to 20 while the state of the art work achieves less throughput of 60 Kbps for the same number of tasks.

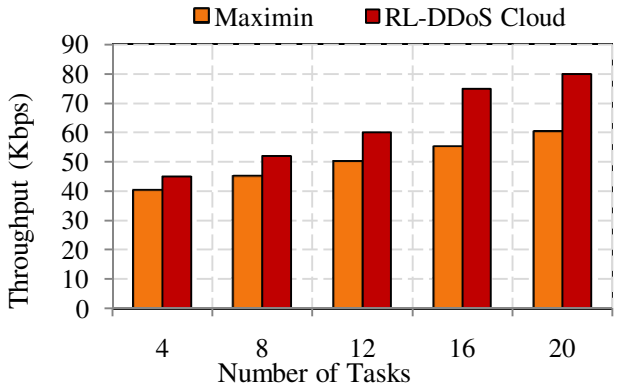

FIGURE 9: Number of tasks Vs Throughput.

4. Resource Utilization Comparison

The number of resources successfully utilized by the proposed model is known as resource utilization rate (R^{uti}). Generally, a good model utilizes more resources in the network. The resource utilization can be formulated as,

$$R^{uti} = Tot_R - Con_R \tag{35}$$

Where, Tot_R is the total number of resources and Con_R is the number of resources that are consumed by the model. The graphical fig 10 shows that, when the number of tasks increases the resources, rate also increases. From our proposed work achieves high resources utilization rate, the reason for the proposed work to achieve high resource utilization rate is that proposed DDoS detection and risk aware VM isolation method. In DDoS detection method, the over utilization of resources is reduced by mitigating the malicious users using fuzzy VIKOR algorithm which classifies the normal and malicious packets based on the multiple criteria. In risk aware VM isolation stage, the users are allocated to optimal VMs using metrics such as VM states (i.e., idle, busy, running), VM size, host size, and CPU utilization rate using SAC that improves the resource utilization rate. On the other hand, the existing work maximin relies only on detecting DDoS attacks using two-fold method such as trust relationship construction based on VM resource utilization such as CPU rate, network bandwidth consumption,and memory utilization rate. However, these resources were not enough to detect DDoS attacks in cloud leads to poor resource utilization rate, and also the existing work constraints the users with limited resource budget which limits the users to consume a smaller number of resources thereby also leads to poor resource utilization rate. The numerical results obtained from the graph shows that the proposed work achieves high resource utilization rate of 80 % when the number of tasks increases to 15 whereas the existing work achieves less resource utilization rate of 60 % for the same number of tasks.

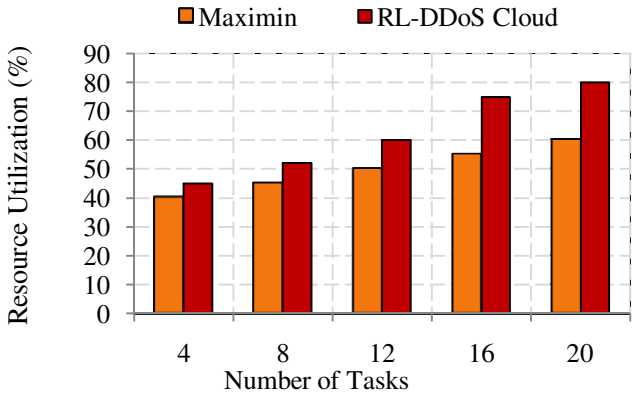

FIGURE 10: Number of tasks Vs Resource utilization.

6.3 Research Summary

This sub-section illustrates the discussion of proposed work RL-DDoS cloud performance in the experimental and results section. From the table IV it is clearly shown that, the proposed work performs better than the existing approach in terms of validation metrics such asresponse time, latency, throughput, and resource utilization. The proposed work found be resist against DDoS attacks in cloud environment by propounding three process. The DDoS detection process analyze the insiders and evaluates their packet, DDoS prevention reduces the illegitimate traffic rate, and VM isolation and migration-based risk reduces the insider's DDoS and outsider DDoS in resilient manner. All the numerical results achieved by the proposed work are higher than the existing works.

Performance Metrics		RL-DDoS Cloud	Maximin
Response Time (ms)	# of tasks	0.5 ± 140	0.5 ± 188
Latency (ms)	# of tasks	0.4 ± 52.2	0.4 ± 75.4
	# of VMs	0.4 ± 30	0.4 ± 64
Throughput (Kbps)	# of tasks	0.3 ± 62.4	0.3 ± 50
Resource utilization (%)	# of tasks	0.2 ± 62.4	0.2 ± 50

TABLE 4: Average Experimental Results.

7. CONCLUSION AND FUTURE WORKS

The security and reliability to the cloud environment is ensured by proposing RL-DDoS Cloud framework. The state of the issues such as poor QoS, and limited security was addressed in this work by enabling end-end to method through three stages. The first stage involves multi factor authentication in which the illegitimate traffics by the mischievous users are pruned out using SHA-512 algorithm that considers three factors such as user ID, password, and random number. The second stage involves DDoS detection in which user data packets are analyzed such as Source Destination IP Address, IP Address, Destination Port, Source Port, Protocol Type and Packet Size using fuzzy VIKOR algorithm which classifies the user packets as normal and malicious. The third stage involves risk aware VM migration and isolation using SAC algorithm in which the VM are migrated based on VM states such as idle, busy, and running, and compromised VMs are migrated by monitoring and considering security constraints. The proposed RL-DDoS Cloud is simulated using Cloudsim simulator tool and performance of the proposed work is validated with several validation metrics.With the proposed research, we achieved better performance in terms of security and reliability over existing works such as response time (140ms), latency (52.2ms with number of tasks, and 30ms with number of VMs), throughput (62.4Kbps), and resource utilization (62.4%).In addition to that, this research also

addressed the major research question of attaining higher throughput rate, and scalability issues by performing distributed and AI based methodology.

The major difference among proposed and existing research was that, none of the existing researches provides end-to-end solution for the DDOS detection and mitigation. The proposed work user initial level security, data security, and cloud security. In addition to that, we have tradeoff both the reliability and QoS in attack detection and user experience by utilizing AI.Our research can be applicable to real timecloud based application we have provide distributed and end-to-end secure framework and also provides considerable results in the simulation results.

In future we plan to adopt blockchain technology to improve the security to cloud computing environment, and also proposing framework which will be suitable for other kind of attacks.

8. REFERENCES

Al-Dulaimy, W. Itani, R. Zantout, & A. Zekri (2018). 'Type-aware virtual machine management for energy efficient cloud data centers', Sustain. *Compute Informatics Syst.*,19, 185-203.

A. Chaturvedi, P. Kumar, K. Sharma (2018). "Analysis on DDoS Attacks and Solutions in Virtualized Cloud Environment". *International Journal of Computer Trends and Technology*, Vol 58, 76-81.

A. Khalid, M. Shahbaz." (2013) cloud computing technology, Department of Computer Science, Government College University Lahore, 1Department of CS & E, UET, Lahore, *Pakistan Journal of Science* (Vol. 65 No. 3 September

A. Sahi, D. Lai, Y. Li, M. Diykh (2017). "An Efficient DDoS TCP Flood Attack Detection and Prevention System in a Cloud Environment, *IEEE Access*.

A. Abhishta, R. Joosten & L.J. Nieuwenhuis.(2017) Comparing Alternatives to Measure the Impact of DDoS Attack Announcements on Target Stock Prices', *J. Wirel. Mob. Networks Ubiquitous Compute Dependable Appl.,* 8, 1- 18

A. Abhishta, R. Joosten & L.J. Nieuwenhuis.(2017) 'Analyzing the Impact of a DDoS Attack Announcement on Victim Stock Prices'.*25th Euromicro International Conference on Parallel Distributed and Network-based Processing (PDP)*, 354-362.

A.K. Lamba, S. Singh, S. Balvinder, N. Dutta & S. Rela. (2018)'Embedding Machine & Deep Learning for Mitigating Security & Privacy Issues in IoT Enabled Devices & Networks. MatSciRN', Other Materials Performance (Topic). *International Journal for Technological Research in Engineering*, ISSN (Online): 2347-4718..

B. Alshehry, W. Allen, (2017). "Proactive Approach for the Prevention of DDoS attacks in Cloud Computing Environments", *Applied Computing and Information Technology*, PP. 119-124

Botvinick, M.M., Ritter, S., Wang, J.X., Kurth-Nelson, Z., Blundell, C., & Hassabis, D. (2019). Reinforcement Learning, Fast and Slow. *Trends in Cognitive Sciences, 23*, 408-422.

Bhardwaj, A., Mangat, V., Vig, R., Halder, S., & Conti, M. (2021). Distributed denial of service attacks in cloud: State-of-the-art of scientific and commercial solutions. *Comput. Sci. Rev., 39*, 100332.

C. Culnane, M. Eldridge, A. Essex & V. Teague (2017). 'Trust Implications of DDoS Protection in Online Elections', *E-VOTE-ID*

D. Ameyed, F. Jaafar, J. Fatahi. (2015) 'A slow read attack using cloud.*7th International*

Conference on Electronics Computers and Artificial Intelligence ECAI) ", SSS-33-SSS-38.

Dong, S., Abbas, K., & Jain, R. (2019). A Survey on Distributed Denial of Service (DDoS) Attacks in SDN and Cloud Computing Environments. *IEEE Access, 7*, 80813-80828.

F. Ma & L. Zhang (2015). 'Multi-objective optimization for dynamic virtual machine management in cloud data center', 2015 *6th IEEE International Conference on Software Engineering and Service Science, (ICSESS),* 170-174

Grady, J. S., Her, M., Moreno, G., Perez, C., & Yelinek, J. (2019). Emotions in storybooks: A comparison of storybooks that represent ethnic and racial groups in the United States. *Psychology of Popular Media Culture, 8*(3), 207–217. https://doi.org/10.1037/ppm0000185.

H. S. Mondal, M.T. Hasan, M.B. Hossain, M.E. Rahaman & R. Hasan. (2017). 'Enhancing secure cloud computing environment by Detecting DDoS attack using fuzzy logic', 2017,*3rd International Conference on Electrical Information and Communication Technology (EICT)*, 1-4, Doi: 10.1049/iet-ifs.2017.0500.

I. Baltatescu (2014) "Cloud Computing Services: Benefits, Risks and Intellectual Property Issues. Global Economic Observer", *IT & Communication Department Romanian Academy 2.*

J. Askhoj, M. Nagamori & S. Sugimoto. (2010)" Archiving as a service: a model for the provision of shared archiving services using cloud computing. '11. Pages 151–158.

K. Singh, K. Thongam & T. De. (2018)" Detection and differentiation of application layer DDoS attack from flash', events using fuzzy-GA computation *IET Inf. Secure.,* Vol. 12 Iss 6, pp. 502-512.

K.A. Simpson, S. Rogers & D.P. (2020) 'Pezaros. Per-Host DDoS Mitigation by Direct-Control Reinforcement', Learning". *IEEE Transactions on Network and Service Management,* 17, 103-117.

Levine, S., Kumar, A., Tucker, G., & Fu, J. (2020). Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems. *ArXiv, abs/2005.01643*.

M. Alkasassbeh (2017). "An empirical evaluation for the intrusion detection features based on machine learning and feature selection methods", ArXiv, abs/1712.09623 .

M. Darwish, A. Ouda, L.F. Capretz. (2013, jun) "Cloud-based DDoS attacks and defenses 'International '*International Conference on Information Society. (i-Society 2013),* 67-71.

M. Idhammad, K. Afdel, & M. Belouch, (2018). 'Detection System of HTTP DDoS Attacks in a Cloud Environment Based on Information Theoretic Entropy and Random Forest', *Secure. Commun. Networks,* 2018, 1263123:1-1263123: 13. /doi.org/10.1155/2018/1263123

M. Mekala & P. Viswanathan. (2019). Energy-efficient virtual machine selection based on resource ranking and resource ranking and utilization factor approach in cloud computing for IoT. Compute. *Electr. Eng*., 73, 227-244.

M. Walterbusch, B. Martens & F. Teuteberg. (2015) "A Decision Model for the Evaluation and Selection of Cloud Computing Services": A First Step Towards a More Sustainable Perspective. Int.

M.Walterbusch, B.Martens, F. Teuteberg (2018). "Evaluating cloud computing services from a total cost of ownership perspective. *Management Research Review*", 36, 613-638.

N. Joshi, S. Shah. (2019). 'A Comprehensive Survey of Services Provided by Prevalent Cloud Computing Environments', *Smart Intelligent Computing and Applications*, Vol104.

O. Yevsieieva & S. M .Helalat (2017). 'Analysis of the impact of the slow HTTP DOS and DDOS attacks on the cloud environment., 2017, *4th International Scientific-Practical Conference Problems of Info communications.*

O. A. Wahab, J. Bentahar, H.Otrok, A. Mourad. (2017) "Optimal Load Distribution for the Detection of VM-based DDoS Attacks in the Cloud", *IEEETransactions on Services Computing*, PP. 1-14.

O. Yigitbasioglu (2014). "Modelling the Intention to Adopt Cloud Computing Services: A Transaction Cost Theory Perspective". *Australas. J. Inf. Syst.*,18(3), pp. 193-210.

P. Sareen. (2013, March) "Cloud Computing: Types, Architecture, Applications, Concerns, Virtualization and Role of IT Governance in Cloud", *Journal of Advanced Research in Computer Science and Software Engineering*, Volume 3, Issue 3, March 2013 pp. 533-538.

R. Bhoyar, N. Chopde (2013, March). "Cloud Computing: Service models, Types, Database and issues". Volume 3, Issue 3.

R. K. Deka, D.K. Bhattacharyya, J.Kalita, (2016) 'DDoS Attacks: Tools, Mitigation Approaches, and Probable Impact on Private Cloud Environment', ArXiv, abs/1710.08628.

S. Sambangi, & L. Gondi. (2020). "A Machine Learning Approach for DDoS (Distributed Denial of Service) Attack Detection Using Multiple Linear Regression.'

S. Vetha, & K. Devi. (2021). "A trust-based hypervisor framework for preventing DDoS attacks in cloud. *Concurrency and Computations',Practice and Experience,* 33. DOI: 10.1002/cpe.5279

T. Hirakawa, K .Ogura, B. B. Bista & T. Takata (2018). 'An Analysis of a Defense Method against Slow HTTP DoS Attack. 2018 International Symposium on Information Theory and Its Applications (ISITA).

T. Hirakawa, K. Ogura, B. B. Bista & T. Takata (2016). 'A Defense Method against Distributed Slow HTTP DoS Attack', 2016,*19th International Conference on Network-Based Information Systems (NBiS).* DOI 10.1109/NBiS. .2016.58.

V. Behzadan & A. Munir. (2018) "The Faults in Our π∗s: Security Issues and Open Challenges in Deep Reinforcement Learning', arXiv:1810.10369v1 .

V. Chang (2018). "An overview, examples, and impacts offered by Emerging Services and Analytics in Cloud Computing virtual reality" Neural Computing and Applications, 29, 1243-1256.

X. Chai, Y. Wang, C. Yan, Y. Zhao, W. Chen & X. Wang. 'DQ-MOTAG: (2020) "Deep Reinforcement Learning- based Moving Target Defense Against DDoS Attacks", *IEEE Fifth International Conference on Data Science, in Cyberspace DSC)*,375-379 DOI:10.1109/DSC50466.2020.00065

Zhang, K., Yang, Z., & Başar, T. (2019). Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms. *ArXiv, abs/1911.10635*.