IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 19, 2020, accepted May 29, 2020, date of publication June 15, 2020, date of current version June 24, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3002333

Novel Three-Tier Intrusion Detection and Prevention System in Software Defined Network

AMIR ALI [1,2] AND MUHAMMAD MURTAZA YOUSAF [1]
[1]Punjab University College of Information Technology, University of the Punjab, Lahore 54000, Pakistan
[2]Department of Statistics and Computer Science, UVAS, Lahore 54000, Pakistan

Corresponding author: Amir Ali (amir.ali@uvas.edu.pk)

ABSTRACT Software Defined Network (SDN) is a flexible paradigm that provides support for a variety of data-intensive applications with real-world smart Internet of Things (IoT) devices. This emerging architecture updates with the managing ability and network control. Still, the benefits are challenging to achieve due to the presence of intruder flow into the network. The research topic of intrusion detection and prevention system (IDPS) has grasped the attention to reduce the effect of intruders. Distributed Denial of Service (DDoS) is a targeted attack that develops malicious traffic is flooded into a particular network device. These intruders also involve even with legitimate network devices, the authenticated device will be compromised to inject malicious traffic. In this paper, we investigate the involvement of intruders in three-Tier IDPS with regard to user validation, packet validation and flow validation. Not all the authentication users can be legitimate, since they are compromised, so that the major contribution is to identify all the compromised devices by knee analysis of the packets. Routers are the edge devices employed in first tier which is responsible to validate the IoT user with RFID tag and encrypted signature. Then the authenticated user's packets are submitted into second tier with switches that validates the packets using type-II fuzzy filtering. Then the key features are extracted from packets and they are classified into normal, suspicious and malicious. The mismatched packets are analyzed in controllers which maintain two queues as suspicious and normal. Then suspicious queue packets are classified and predicted using deep learning method. The proposed work is experimented in OMNeT++ environment and the performances are evaluated in terms of intruder Detection Rate, Failure Rate, Delay, Throughput and Traffic Load.

INDEX TERMS SDN security, IoT, intrusion prevention system, RFID, packet classification.

I. INTRODUCTION

Software Defined Network (SDN) is an underlying infrastructure developed to distribute a wide range of traffic into the network. SDN with Internet of Things (IoT) meets the requirements of different applications by monitoring the arrived traffic. This is attained by the separation of data plane and control plane. Data plane with switches are composed with flow tables using which the packets are matched and forwarded as per the action [1]–[3]. This notably explores SDN with IoT where thousands of devices are involved. The combination of SDN and IoT is subjected to vulnerable threats [4]–[6]. Security has become one of the serious bottleneck issue that needs to be solved. The common requirements of security are privacy, confidentiality, integrity and control access. Security is compromised while the attackers increase and they cause scarcity of network resources and bandwidth. The abnormal traffic into the network is suspected to be uncommon that obviously degrades network performances.

Security policies are presented to control the traffic flow into the network in [7]–[10]. The arrived traffic is classified according to the designed flow rules. While a new flow is arrived, then the particular packet is forwarded to next layer for analysis. Monitoring the flows strengthen security in the network environment. The conventional rules are also being forged in recent days due to the intelligence of the intruders into the network. The access to flow traffic is allowed only when the entire flow entry matches. The anomaly flows into the network by the intruders is detected and mitigated [11], [12]. Machine learning algorithms as Decision Tree, K-Nearest Neighbors (KNN), Naïve Bayes, Neural Network, Support Vector Machine (SVM) and Self-Organizing

The associate editor coordinating the review of this manuscript and approving it for publication was Xujie Li.

Map (SOM) are used to analyze the packet features that exist in the flow. The utilization of machine learning improves accurate detection of intruders. Analyzing the network flows ensures to mitigate intruders and especially the flooding of excess flows is performed by Distributed-Denial-of-Service (DDoS) attack [13], [14]. DDoS attacks are serious threats in SDN environment which continuously increases problems for the legitimate users. The security measures is essential in SDN based IoT environment. In order to prevent the attackers, authentication is a promising solution [15]. The IoT users have to register with the administrator for accessing the network. In this way, the registered users are only provisioning with the permission to access the network. Also, the security credentials are considered to be unique and also required to manage the credentials using cryptography. In recent days the network environment is surrounded by intelligent intruders to degrade network environment performance.

The key research questions identified in SDN-IoT with the security aspects are:

(i) How to mitigate the compromised users in the network?

(ii)How to discover vulnerable threats that are participating in the network?

In this paper the above two research questions are addressed in proposed three-Tier SDN-IoT architecture. The designed architecture concentrates on resolving the security concerns with the processing of validating users, packets and flow. The individual user is authenticated with RFID and signature. Also the security credentials are transmitted secure to mitigate man-in-middle-attack (MIMA). Then packets are validated with the features in switches of data plane and suspicious flow is validated in control plane. The three-Tier architecture addresses a secure environment that detects intruders and takes steps to prevent the intruders by authentication and flow rule update in data plane. This section further discusses the motivation, contribution and organization of the paper.

A. MOTIVATION

Security is one of the most challenging issue that is to be concentrated for improving the effective utilization of the network resources and traffic load. Internet is a basic requirement to perform day-to-day activities and hence the traffic flow into networks is significantly increased. Hereby the focus on traffic load balancing presents with the idea of reducing the malicious traffic which will certainly improve scalability. This issue is handled in real-world environment and so this is motivated to be resolved. The motivation towards security has developed three-Tier architecture in SDN-IoT environment to provoke a scalability supported system. Even though the intelligence of attackers are grown, this work detects and prevents intruders by analyzing the individual's credentials, packet features and flow using artificial intelligence and machine learning algorithm. The arrival of packets are in large number and so these faster operating and decision making algorithms are preferred to detect and prevent intruders.

B. CONTRIBUTION

SDN-IoT environment is constructed into a three-Tier architecture concentrated on mitigating the attacks that reflects in degrading the network performance. The common security issue due to intruders is taken in account and possible solutions are defined. Even though the users are authenticated, some of the users are compromised and hence they are detected by packet analysis in this work. This paper investigates the security concerns in SDN-IoT environment and the major three-fold contributions are:

- To mitigate the participation of illegitimate IoT users, authentication is performed by validating the user based on the RFID identity and signature. A symmetric algorithm is proposed for faster and efficient encryption of the security credentials.
- In order to predict the compromised user packet the OpenFlow switches are deployed with a new filtration using Type-2 fuzzy from the extracted features and categorize the packets into normal, suspicious and malicious. The malicious packets are discarded, then the suspicious packets are forwarded to control plane.
- Not all the packets could be analyzed at the single stage and so lastly the suspicious packets are analyzed with deep learning for appropriate prediction of intruder packet and identify the source device.
- The experimental performances of this proposed system improves significant parameters such as Detection Rate, Failure Rate, Delay, Throughput and Traffic Load.

C. LAYOUT OF THIS PAPER

This paper is organized into set of sections for elaborating the security study. The composition of this paper is as follows, Section II details all the recent review on SDN security, Section III illustrates the problem defined from the area of SDN intrusion detection, Section IV explains the proposed solutions that solves security problem, Section V is the demonstration of justification for the efficiency and improvements achieved for the proposed solutions and finally Section VI illustrates the conclusion and future direction of this research.

II. RELATED WORK

In this section the existing research works are studied to identify the existence of intrusion detection system (IDS) in SDN-IoT environment. Most of the works have been developed to detect intruders whereas the prevention is less focused. A reliable security-oriented routing mechanism namely Route Guardian was proposed [16]. This mechanism was designed in SDN architecture that comprised of policy parser, resource status monitor, routing rule generator and incident reactor. The switches receive requests from hosts, and then policy was created by policy parser, later it was given to resource status monitor module. This module was equipped to periodically aggregate network resource based metrics. If any malicious flow with new packet features, then it was

allowed into network. The attackers were detected using security policies that were defined with packet features [17]. These policies are generated at controllers. The intrusions in the network are attackers that involve for demolishing the network performance.

In SDN-IoT an intelligent detection of anomalies i.e. malicious traffic was presented [18]. Machine learning algorithms were used in IDS which improves the accuracy in detection. The algorithms SVM, SOM and Stacked Auto encoder Deep Learning approach-SAE were used in IDS over individual network entities. These algorithms were higher in detection accuracy but as per the increase in arrival rate of attacks, their accuracy degrades. Then a hybrid machine learning model was designed for detecting DDoS attacks [19]. The algorithms used in this work are SVM and SOM. The flows from the switches are collects and processing in control plane which was deployed with SVM, SOM and enhanced history-based IP filtering scheme (eHIPF). In SVM and SOM, the flow duration, packet number, byte number and protocol were used to classify the packets and process in eHIPF. This eHIPF scheme extracts time, packet per flow, priority, flow number, protocol and flag based on which the abnormal source was detected. The poor selection of key parameters in SVM leads to degrade classification performance and SOM is difficult to determine the map size and processed well only when the training data is larger. In [20], Artificial Neural Network (ANN) architecture was used for classifying the benign and malicious network traffic. The arrived packets were processed in ANN; here the key demerit was the number of neurons which reduces accuracy. The ANN was not able to support for large scale network and hence it consumes higher processing time. Deep Neural Network (DNN) was built to detect the anomalies based on the flow [21]. The packet features are extracted and then they were classified. The use of single controller causes single point failure due to multiple processing by the same controller.

The participation of attackers has to be detected as early as possible which was studied in [22]. For earlier detection a set of rules were defined based on which, the IDS block drops the bad flow and processes the good flow. Even though the IDS detect bad flows earlier, it is performed at controller which means the bad flows are allowed into switches which may be affected priory. Hence the need for intrusion detection and prevention system (IDPS) was increased to detect and mitigate the activities of malicious flows [23]. In this work, two connection based techniques were proposed which are credit-based threshold random walk (CB-TRW) and rate limiting (RL). The port scanning attacks are overwhelmed by verifying the port numbers of the received packets. Initially to prevent the intruders, the packets in blacklist were verified. Then for intrusion detection the packet features as protocol, flag and counter values are validated. The prevention of intruder with the blacklist was not sufficient since, the intruder will also learn the environment and approach network with new matching packet flows.

In [24], [25], Recurrent Neural Network (RNN) was applied for detecting DDoS attacks with tuned parameters. The learning parameters are tuned based on the features and then applied for attack prediction. Extreme Learning Machine (ELM) was also incorporated for attack detection [26]. The ELM offers to select packet features for attack prediction whereas the significant packet features as IP address, port number may be ignored. Multilayer Perceptron (MLP) Model was developed which processes the extracted features [27]. Also optimization algorithms were presented for accurate prediction of attack traffic [28]. Lion optimization algorithm was used for selecting optimal set of features for detecting DDoS attack traffic. After selection of features, Convolutional Neural Network (CNN) classifies the packets. The DDoS attack was also detected using pattern graph model that autonomously detects attack [29]. The packets collected from switches are processed in controllers by graphs and the model was periodically updated. This work was not able to detect assist new flow and also it requires timely update. To block attackers into the network, authentication was developed by which the individual's unique credentials are validated [30]–[33]. Chaotic secure hashing in combination with a digital signature was used to authenticate user and then the user packets are analyzed. Rivest-Shamir-Adleman (RSA) algorithm was involved for secure transmission of the user credentials to the verifier entity [34]. The verification was provided by validating security parameters, device identity and password. These constraints were easily leaked and guessed by attackers, hence poor verification. In [35], the IDS monitors and as well as it authenticates by signature. In this work, the signature was generated based on the packets. Hereby it also generates signatures for malicious traffic. Attacks are majorly detected with IP address in which the malicious packets are maintained in blacklist [36].

III. PROBLEM DESCRIPTION

This section summarizes the specific problems on SDN-IDS. A multiple IDS were proposed for faster intrusion detection [37]. This algorithm determines the close relationship in terms of routers path and groups. Based on the similarity in flow paths, the flows are grouped using Principal Component Analysis (PCA) and Gravity-based clustering balance the load. The problems in this work are data rate of each flow differs with each other, so balancing the load using this metric is unfair. The fluctuating group size fails to balance traffic flow. Intruders were allowed into the system reduces network performance. Then an entropy-based lightweight DDoS flooding attack detection model was proposed with two attributes as destination IP address and destination port [38]. All the edge switches estimate entropy to identify anomaly. However the entropy prediction is dynamic by considering two packet features which is not sufficient for accurate attack detection. These two packets features are conventional and hence the variation in entropy will not vary often but which consumes resource for repeated computation. OpenFlow Security (OpenSec) framework was developed for implementing

TABLE 1. Existing problems.

Method/Technique	Concept	Drawback
Multiple IDS [37]	Grouping of arrived flows in controllers for detection	Dynamic grouping allows malicious traffic. Flow data rate can be varied.
Entropy Based Method [38]	Dynamic threshold using IP address and port number	Frequent computation degrades network performance. Intermediate switches can be attacked so easily.
OpenSec Framework [39]	Human Readable Policies	Self-definable policies allow intruders to add own policies.
Multi-Queue Method [40]	Maintenance of each queue for each switch	Singly queue will be filled due to larger traffic in particular area.

security policies that are human readable [39]. In OpenSec framework a matching pattern was specified in policy specification language component for matching the flow, service and it reacts accordingly. In this work, the end-user defines policy, so even a malicious user can add policy according to their requirement. This allows malicious flows into the network. Human-readable policies are unfair, since purposely some person could use/modify the policy for their convenient. The intensity of DDoS attacks was reduced with the idea of multi-queue SDN controller scheduling algorithm based on time slice allocation strategy was proposed [40]. The logical queue was executed based on polling mode which has nil DDoS attacks in it. The individual queue was maintained for each switch, as per the increase in number of switches the number of queues also increases (Difficult in queue maintenance) and if a switch does not receive any flow or receives less flow, there the queue deployment and resource are wasted. Here, the controller needs to verify each flow in the queues one after the other, which consumes larger time. This may lead to severe damage to switches and network elements.

The problem statements of these researches are the use of self-defined policies, multi-queues and flow verification that were not effective in detecting attacks and also most of the research focus is to only detection. The prevention of attacks will gradually increase network performance. The problems stated are illustrated in Table 1. In this way, our proposed system is carried out towards the solutions to detect and prevent intruders and shows significant improvisation.

IV. PROPOSED IDPS SYSTEM

The proposed system addresses the challenging issue of security in SDN-IoT environment. The Internet of Things (IoT) devices are becoming more popular in wide variety of applications that eventually increases intruders. This section is categorized into sub-sections with the development of IDPS system to mitigate malicious traffic flows. The validation is carried out in three layers to ensure strong protection in the designed SDN-IoT environment.

TABLE 2. Notations guide.

Notations	Description
U	IoT User
T_g	RFID tag
TU_{id}	Identity of RFID tag
T_R	Random number of tag
$\|\|$	Concatenation
$\oplus$	XOR operation
ID	User's device identity
P_k	Private key from prime number
TA	Trusted Authority
H	Hash
p, q	Prime numbers

Table 2 enlists the notations used in this paper along with its short description.

A. THREE-TIER SDN-IoT ARCHITECTURE

A three-Tier SDN-IoT Architecture is designed in three consecutive layers with network entities. The processing handled in each Tier is user validation, packet validation and flow validation. Initially the malicious traffic is generated from the IoT users with the intention to demolish the performance of the network. The entities and the process presented in each layer are as follows,

Tier 1– In this Tier, there exist participation of both legitimate and illegitimate IoT users. Let the IoT users be $\{U_1, U_2, U_3, \ldots.., U_i\}$ in which few are illegitimate i.e. present without original security credentials. The IoT users submit traffic into the edge devices i.e. gateways represented as $\{G_1, G_2, \ldots\}$. The security credentials of the legitimate users are managed at Trusted Authority (TA).

Tier 2– In this Tier, OpenFlow switches are deployed as $\{s_1, s_2, s_3, \ldots.s_n\}$. Each switch maintains a flow table with administrator defined rules that takes action using the algorithm.

Tier 3– In this Tier, the N number of distributed controllers $\{C_1, C_2, C_3, \ldots., C_N\}$ are presented with the maintenance of two separate queues. Controller is responsible in validating the user flows and then it installs new rules into Tier 2 if required.

On designing three-Tier SDN-IoT architecture, the malicious packets from IoT users are detected. User validation is performed to prevent intruders by validating the security credentials. The identity of RFID, signature is verified; here SHA-256 is used for signature generation. Then packets are validated with packet features in type-2 fuzzy logic and lastly deep learning CNN is involved for analysis of suspicious packets. Figure 1 shows the proposed three-Tier SDN-IoT architecture and the processes on each Tier. The processing of each Tier is completely focused on mitigation of the attack packets into the network. The registered users are legitimate, but still there exits malicious packets since the legitimate users are compromised by attackers in the network. As per this work, the compromised legitimate user produces flooding of traffic into the network.

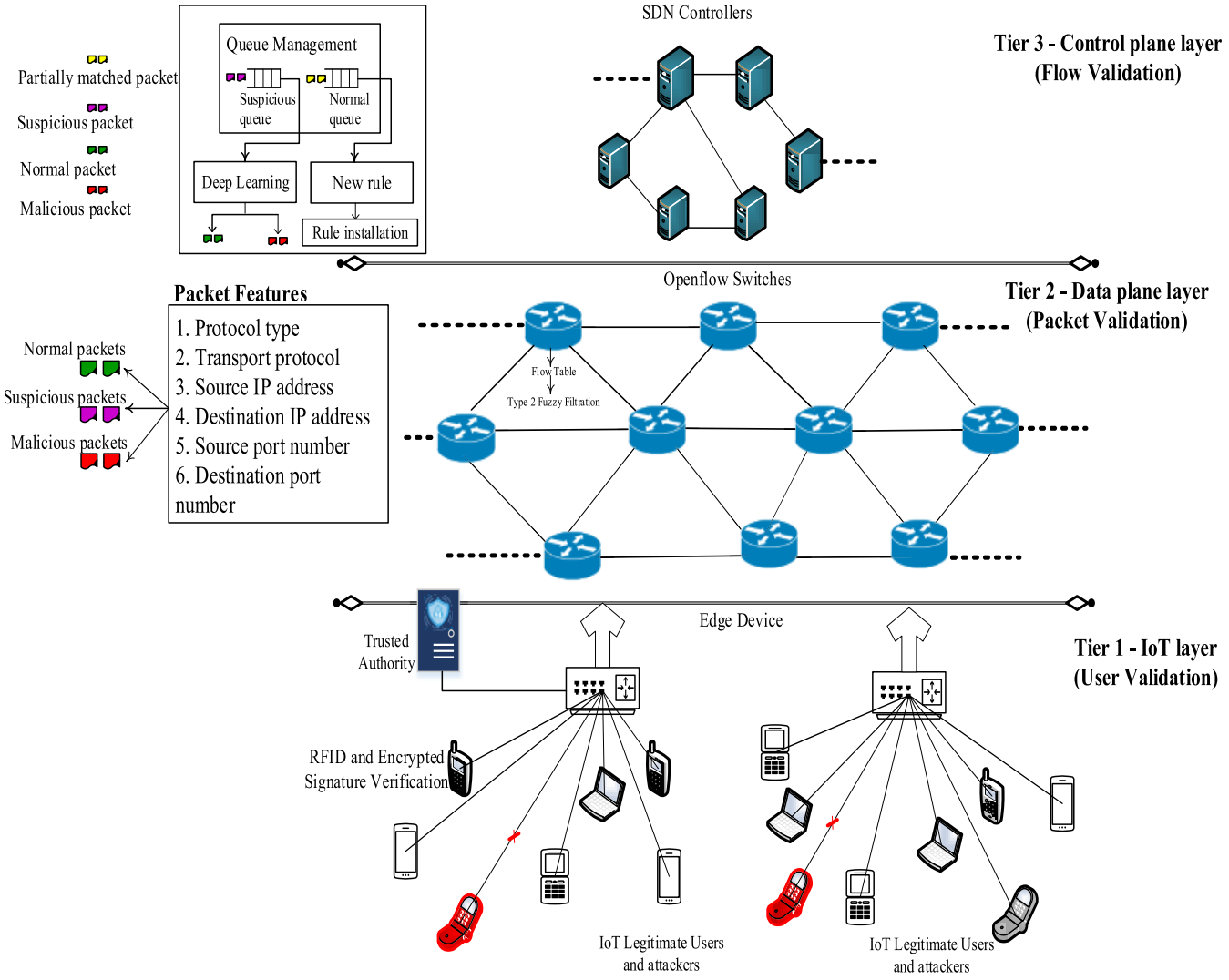

FIGURE 1. Proposed Novel three-Tier IDPS Model.

B. TIER 1 –USER VALIDATION

In Tier 1 the U_i IoT users are validated when the user requests with a service. User validation includes processing of two phases as (I) Registration phase and (II) Authentication phase. The steps in registration phase are given as follows,

STEP 1: First assume U_1 be the IoT user who requests the TA for registration with his / her unique T_g. Along with this each user submits $TU_{id(1)}$ and ID to the TA for registration.

$$U_1(TU_{id(1)}||ID) \rightarrow TA \tag{1}$$

STEP 2: Then on receiving the $TU_{id(1)}$ from U_1, then the TA generates a T_R for each user tag and a unique signature using SHA-256 algorithm. For signature generation, select a k random value ranging between $[1 \ldots\ldots q-1]$, and then compute $r = \left(g^k modp\right) modq$. Further compute $s = \left(k^{-1}\left(H\left(TU_{id}\right) + (P_k)r\right)\right) modq$. Here g is a generator. The generated signature (r, s) is given to U_1. If $r = 0$ and $s = 0$, then select a new random number k. Then the generated signature is delivered to corresponding IoT user here it is U_1.

$$TA \rightarrow U_1(r, s) \tag{2}$$

STEP 3: Later the signature from TA is stored in user's device which is used during the time of authentication. With this step, the registration of the device is completed.

The steps followed in Authentication Phase are depicted below:

STEP 1: During authentication the request from U_1 is submitted to G_1 with ID. On receiving request, the timestamp T_1 freshness is verified. Then the G_1 verifies user's ID and it asks for TU_{id} of the corresponding user.

$$U_1(ID) \rightarrow G_1 \tag{3}$$

STEP 2: Upon receiving the response from G_1 the timestamp freshness is verified as $(T_2 - T_1) \leq \Delta T$. Only if the timestamp exists the authentication will be proceeded. Then U_1 computes $K = (T_R \oplus TU_{id(1)})$ and sends with

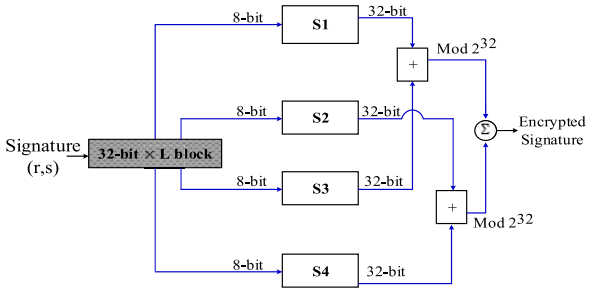

FIGURE 2. Modified BlowFish Algorithm.

timestamp T_3.

$$U_1(K = (T_R \oplus TU_{id(1)})) \rightarrow TA \tag{4}$$

STEP 3: Next the TA receives K from U_1 and verifies timestamp $(T_3 - T_2) \leq \Delta T$.Then validates $TU_{id(1)}$ and then validation for signature is invited.

$$TA(invites) \rightarrow U_1(r, s) \tag{5}$$

STEP 4: The U_1 receives the invite message and then encrypts its own signature (r, s) using modified blowfish algorithm. Before encryption, the $(T_3 - T_2) \leq \Delta T$ is verified. The encrypted security credentials enables to resist from attackers and also the privacy of security credentials is hard to be leaked. Symmetric encryption is used in order to reduce the complexity in key generation.

This algorithm is a symmetric block cipher which is faster in encrypting the signature. The signature (r, s) is a 64-bit block with 32-bit key for encryption. The feistal network structure is used. In this modified blowfish algorithm three steps are followed as generation of sub-keys, substitution boxes and encryption. The proposed encryption uses four substitution boxes as $S[0], S[1], S[3], S[4]\}$ as shown in Figure 2. The encrypted signature is sent to TA for verification.

$$U_1E(r, s) \rightarrow TA \tag{6}$$

STEP 5: The TA receives the cipher signature in which first it checks timestamp $(T_4 - T_3) \leq \Delta T$. If the timestamp exists, then signature is decrypted and verified. In case if the signature is not true, then the user is not allowed. If true, then the access notification is provided to the gateway.

STEP 6: Only after receiving notification access from TA, the gateway receives packets from the users and forwards to switches.

The authentication of users enables to withhold the illegitimate user's access into the network. This authentication is performed since; most of the attack packets are arrived from illegitimate users. A system model without authentication is much easier for the attackers to launch the attack to degrade network performance. In order to avoid illegitimate users, authentication is performed [41]–[44].

The Tier-1 process of user validation is illustrated in Figure 3, using which the individual user is identified to be

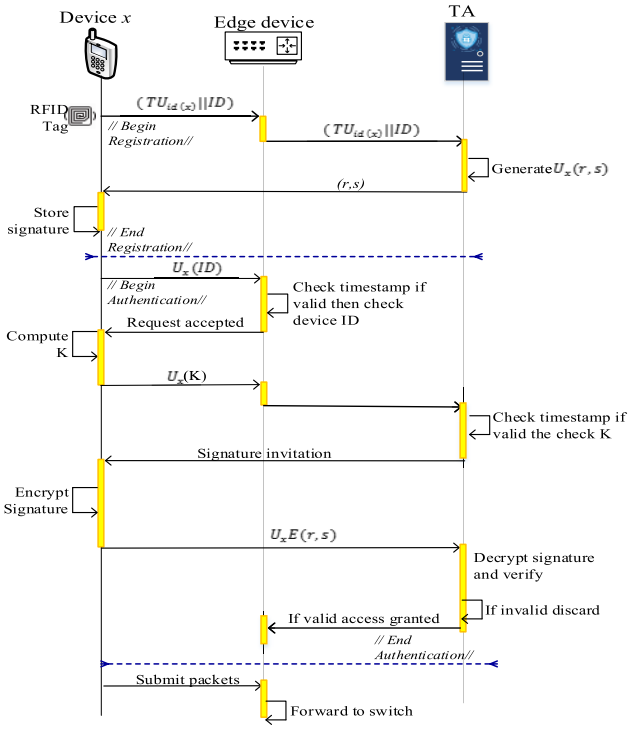

FIGURE 3. User Validation in Tier – 1.

original or fake. The authentication of user by RFID and signature ensures to allow access only for legitimate users. The goal of our work is to detect and prevent intruders. This authentication is also provided prevention which substantially resists illegitimate users into the network.

C. TIER 2 –PACKET VALIDATION

The packets from the legitimate users enter into Tier 2 of OpenFlow switches where the packets are validated using features. Even though the intruders are prevented at Tier 1, the compromised user's participation exists in the network. In this work, the compromised users will flood malicious packets into the network and drain out the resource of network entities. The arrived packets are validated using type-2 fuzzy filtering in which membership functions are applied. Type-2 fuzzy is capable to process with linguistic uncertainties [45]. Fuzzy system is IF-THEN rules systems which are operated on Fuzzifier, Inference Engine, Type-Reducer and Defuzzifier. In this work, fuzzy logic is applied for processing the membership functions. The key packet features that are extracted from the packet header are Protocol Type, Transport Protocol, Source IP Address, Destination IP Address, Source Port and Destination Port.

As insisted in Table 3, the packet features are processed into fuzzy logic. In this Type-2 Fuzzy, for each primary membership function there is a secondary membership function. Initially the Fuzzifier receives the input i.e. packet features and maps it with numeric vector. This vector is the values of the packet features. The rule defined in type-2 fuzzy is similar as in first introduced fuzzy logic. The received multiple inputs

TABLE 3. Notations packet features and description.

Feature	Feature Representation	Feature Description	Example
Protocol Type	pr_ty	Defines the protocol used	HTTP,FTP
Transport Protocol	tr_pr	Used transport protocol	TCP
Source IP Address	src_IP	IP address of source device	*xxx.xxx.x.xx /xx*
Destination IP Address	dst_IP	IP address of destination device	*xxx.xxx.x.xx /xx*
Source Port Number	src_prt	Port number of source device	*yy/yyy*
Destination Port NFumber	dst_prt	Port number of destination device	*yy/yyy*

TABLE 4. Fuzzy rules.

pr_ty	tr_pr	src_IP	dst_IP	src_prt	dst_prt	Output
High	High	High	High	High	High	High
High	High	Low	Low	Low	Low	Low
Low	Low	High	High	Low	High	Medium
High	High	High	Low	Low	Low	Medium
Low	Low	Low	Low	Low	Low	Low
⋮	⋮	⋮	⋮	⋮	⋮	⋮

from the user packets are validated with the rules and then it produces single output. The fuzzy rules are defined based on the correctness of the packet feature. By knowing the correctness of the feature it will be as 'HIGH' else 'LOW'.

The rules are defined from simple IF .... THEN structure which is faster in decision making. The rules in type 2 fuzzy are developed as depicted in Table 4. For instance a set of three rules are defined, in this way rules in fuzzy are defined and packets are validated. Since the type-1 fuzzy is noisy and uncertain, this type-2 fuzzy is used which is capable in creating membership function by its own [45]. Assume X as universe of discourse, the unit interval $U = [0, 1]$. Using this type-1 fuzzy set $F1$ in the inference engine is given as:

$$F1 = \{(x, \mu_{F1}(x))|\mu_{F1}(x) \in U \forall x \in X\} \tag{7}$$

The $F1$ for X that is defined with a membership function of $\mu_{F1} : X \rightarrow U$. Hereby the term U denotes the elements present in set X and u is the membership grades of U. The in type-reducer, interval is determined. Let $F2$ be the type-2 fuzzy set for X, here the reducer $\tilde{F}$ is expressed as:

$$\tilde{F} = \{(x, (u, 1))|\forall x \in X \Lambda \forall u \in J_x \subseteq U\} \tag{8}$$

Let assume $\tilde{F}$ as the fuzzy set and $J_x = \{u \in U|\mu_{\tilde{F}}(x)(u) > 0$. As a result, the secondary membership function is formulated as shown below:

$$\mu_{\tilde{F}_{(x)}}(u) = \begin{cases} 0, & \mu_{\tilde{F}}(x)(u) < z \\ z, & \mu_{\tilde{F}}(x)(u) \geq z \end{cases} \tag{9}$$

In this Type-2 Fuzzy set, it is defined to be unit cube whose surface is represented as (x, u, z) which are the coordinates.

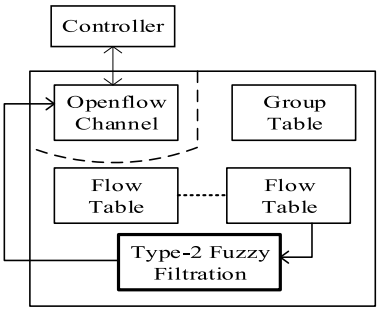

FIGURE 4. Components in OpenFlow Switch.

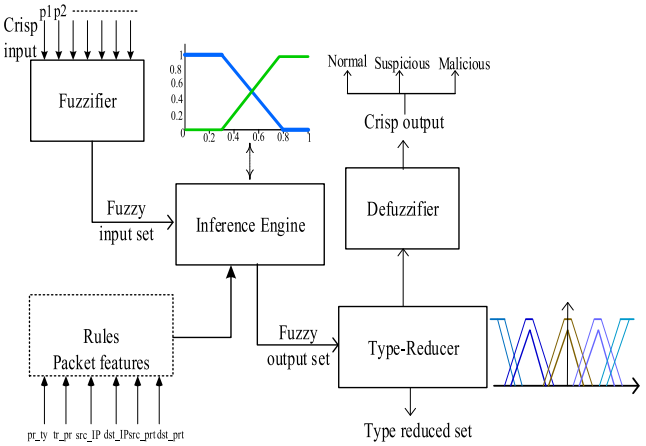

FIGURE 5. Packet Validation in Tier – 2.

This expands as $x \in U, u \in J_x \subseteq U$ and the z-axis is gives as follows:

$$z = \mu_{\tilde{F}}(x)(u) \in U \tag{10}$$

By the prediction of fuzzy rules in primary and secondary level, the decision is made for each packet. The operations in type-2 fuzzy are performed with the defined membership functions. The incoming packets from legitimated users are filtered and hence the consumption of excess bandwidth is reduced. Bandwidth is one of the essential resource to process the arrived packets. Hence, shortage in bandwidth leads to increase packet drop.

The Type Reducer in this fuzzy is used to generate a set of output that is transformed into numeric output and then executed in the Defuzzifier. The modified structure of OpenFlow switch is shown in Figure 4. An additional block for packet filtration is included into switches. The flow table entries are operated based on the instructions. The fields that are used in matching are ingress port and packet headers. Once the action is made the packet will be dropped or processing into filtration. Figure 5 depicts the validation of packet which matches the packet fields and makes action correspondingly.

The incoming packets matches with the fields in flow table and then the features are extracted for filtration process. The filtration significantly maps the key packet features from individual user. The performance of filtration in switches enables to avoid flooding of malicious traffic at initial stage and hence

it is not allowed in controllers. Let the packets from users be $\{p_1, p_2, p_3, \ldots.\}$. from which the features are extracted and converted into crisp values. Initially the fuzzy rules are written using the six features and then type-reducer generates secondary trapezoidal membership functions. Using the type-reduced set, the Defuzzifier filters the packet into normal, suspicious and malicious. The normal packets are forwarded, malicious packets are discarded and suspicious packet will be sent to Tier 3.

D. TIER 3 –FLOW VALIDATION

In this Tier, the suspicious packet flow is validated using Convolutional Neural Network (CNN). The overflow of the packets in the queue is also happens due to the involvement of attacks. The controller maintains two queues commonly for all the switches that are deployed in the network. Before validating the flow, the bandwidth consumption in Tier 2 is measured. Since, bandwidth is one of the significant resource which is excessively occupied by attackers and not allowed for normal packets. The two queues in SDN controller are:

(I) NORMAL QUEUE– The packets that are new to the network are arrived into normal queue. The packet which could not be recognized from the flow rule of the switches will be processed in this queue.

(II) SUSPICIOUS QUEUE– The packets that are partially matched with the fuzzy rule filtration is processed in this queue. The flows of these packets are validated in this Tier.

The packets in normal queue will be analyzed and new rules will be created and installed into Tier 2 switches. On the other hand, the packets in suspicious queue are extracted with in port and out port number. All the arrived suspicious packets are processed one at a time using deep learning. The CNN has the ability to process with large number of input features and hence CNN is preferred to validate flow in Tier 3. CNN is composed of three layers as input, hidden and output [46]. For each packet the in port and out port are extracted and the weight value is estimated in hidden layer. The number of neuron nodes in the hidden layer is lesser than the input layer. Then, the packets are validated using the logistic function in output layer i.e. Softmax Layer. In hidden layer, the weight is computed based on the activation function that depends on the packet features.

Initially a set of packets based on the above six features are collected and trained in the CNN. While the packets arrive during testing it matches with those trained features and then it classifies the packet either suspicious or normal.

V. SIMULATION EVALUATION

This section discusses the experimental environment of the proposed system model. In this section the simulation configurations, comparative analysis and security analysis are studied. The justification for the better performance of proposed three-Tier IDPS is developed.

TABLE 5. Fuzzy simulation parameters.

Parameters	Value
Network Model	
Network area	800 × 800 m
Number of users	6 (Minimum)
Number of attackers	5
Number of edge devices	5
Number of switches	5
Number of controllers	3
Number of TA	1
Packet Model	
Packet interval	2 s
Number of packets	2000
Flow timeout	2s
Service time	0.0098ms
Delay	1 μs
Other Network Configurations	
Data rate	300 Mbps
Link bandwidth	5 Mbps
Transmission range	80 m
Simulation time	300 s

TABLE 6. Specification of RFID tags.

Feature	Applicability
Readable Memory	Present
Coverage distance	~5m
Economical	Yes
Compatible	Yes
Lifespan	High

A. SIMULATION SETUP

The proposed IDPS is implemented using network simulator environment OMNeT++ 4.6. This OMNeT++ is a discrete event simulator that is extensible with the support of all advanced technologies in the field of network. OMNeT++ network simulating tool is installed in windows operating system (OS) which is a commonly used OS. The initialization setup is performed in windows 7 ultimate. At first install JDK 1.8 and then install open source OMNeT++ 4.6 version. After completion of installation, this simulator can be launched and developed with multiple functionalities.

OMNeT++ is suitable for designing SDN network model, since SDN is different from traditional network. In general SDN is designed with OpenFlow switches and controllers to analyze and process arrived packets. In this simulator, flow table module is used to construct flow table entries in individual OpenFlow switch based on which arrived packet is either dropped or forwarded. Hence OpenFlow switch consists of flow table, group table and an OpenFlow channel to forward packets. According to this work, the flow table matching is performed with type2 fuzzy filtration.

The simulation parameter that are used to design this proposed system are depicted in Table 5 and consecutively Table 6 depicts the potentialities of using RFID in this system. By using these parameters we have developed a SDN-IoT network environment as IDPS. The processing of user validation, packet validation and flow validation using algorithms

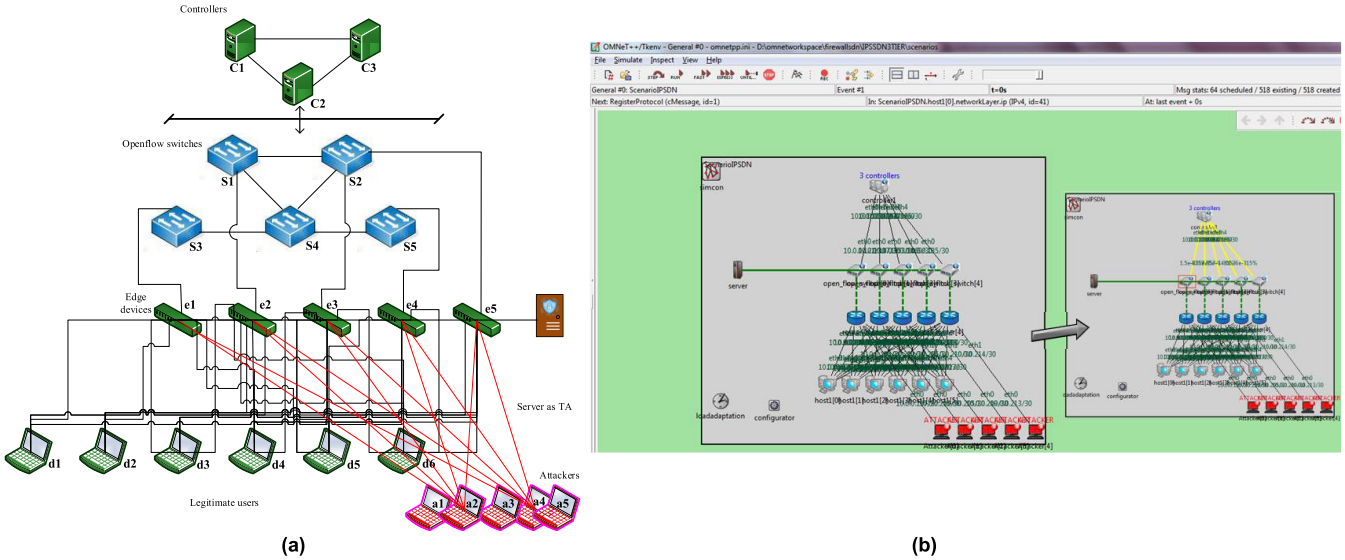

FIGURE 6. SDN-IoT IDPS Environment (a) Network Setup and (b) OMNeT++ Simulation Setup.

are appropriately developed and executed. The simulation parameters are not limited to this set; it is also extended with all the other default network settings.

RFID is an auto identification and data capture technology which enables to recognize the particular using a wireless card. The security credentials are stored in RFID tags and the information are read and transmitted by RFID reader. RFID is one of the promising solution for security in a reputed organization. This work is implemented using simulator and hence the RFID are used as unique identities for individual users those are able to access the network services. This three-Tier IDPS system deals with RFID authentication using unique identity and signature. After user validation the switches takes responsibility to validate the packet followed by flow validation in controllers. This three-Tier processing is carried on SDN-IoT environment with the involvement of illegitimate users as attackers and legitimate users as compromised users who submits continuous traffic into the network. The packet flooding leads to drain the resources of network entities that certainly slow down the network and cause packet drop.

The proposed three-Tier IDPS environment is designed with attackers $\{a1, a2, a3, a4, a5\}$, IoT user devices $\{d1, d2, d3, d4, d5, d6\}$, edge devices $\{e1, e2, e3, e4, e5\}$, OpenFlow switches $\{s1, s2, s3, s4, s5\}$ and controllers $\{c1, c2, c3\}$. The construction of these network entities is shown in Fig. 6. The initial development of the proposed three-Tier IDPS is shown in Figure 6.(a) and using these specifications a simulation environment is created in OMNeT++ which is incorporated with the processing of user validation, packet validation and flow validation.

The developed three-Tier IDPS network environment is applicable for an organization. An organization is a system which has bunches of employees operating the system. Here, the individual employee requires performing RFID

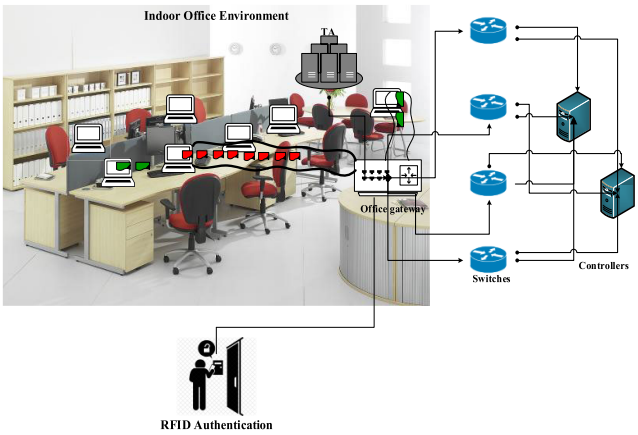

FIGURE 7. Application Scenario.

authentication and hence each user will be provided with unique RFID tag. After authentication the packet validation and flow validation is carried out to limit the malicious packets. This real-time scenario support for three-Tier IDPS system is shown in Figure 7. Apart from this application, the proposed network can also be applied on many other applications.

B. COMPARATIVE ANALYSIS

In this section, the proposed three-Tier IDPS environment is evaluated with the previous research works that are used for attack mitigation as OpenSec Framework and Multi- Queue Method. The experimentations through graphical plots in this section illustrate the significant performances of our proposed system when compared with previous work. The goal of this system is to detect and mitigate the attackers that exist in the network; hence the parameters preferred in this work are detection rate, failure rate, delay, traffic load, throughput, precision and accuracy. The parameters delay, throughput and

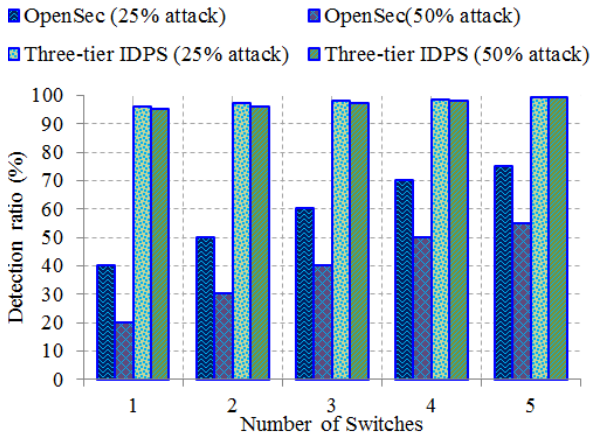

FIGURE 8. Comparison on Detection Rate.

traffic load also impacts the network performance while there is attack participation. As mentioned above, the metrics are evaluated and their comparative results are shown below.

1) DETECTION RATE AND FAILURE RATE

Detection rate is one of the significant parameter that defines the effective prediction of attacks. In three-Tier IDPS the attack packets are detected by validating the packet features which is the main clue to exactly identify the behavior of attacker. In existing OpenSec framework, the attacks are predicted by deep packet inspection (DPI) in which a set of policies are defined by user with which the packets are identified as normal or malicious. Here the detection rate will be low, since the fake policies could be included with that the malicious packets also pass DPI. This detection rate has to be higher even under the increase in the packet density.

The increase in detection ratio ensures with the appropriate processing of the network design for attack detection. The evaluated results for detection rate are depicted in Fig. 8, from which the three-Tier IDPS has higher detection rate at increasing attack intensity. This comparison shows the detection of attack packets with the increase in number of switches.

Here the 50% attack denotes that nearly half of the arrived packets are malicious. As a result, the proposed three-Tier IDPS achieves higher detection rate at increase in attack rate, this increase is due to the appropriate detection of attacks using type-2 fuzzy in switches. Most of the attackers are ignored initially by the RFID authentication; however the attacks packets from compromised users are also effectively predicted in switches.

In this work, the type-1 fuzzy membership functions are designed using the six features followed by type-2 membership function. In comparison, the attack packets are eliminated in previous work with the maintenance of individual queues for each switch. Also the attack is detected only at controller and hence the performance at switches against the attacks is poor. Due to these limitations, the packets are validated at switches by extracting their significant features.

TABLE 7. Average detection rate.

Existence of Attack	Work	Detection Rate (%)
25% of attack	OpenSec	59
	Three-Tier IDPS	97.72
50% of attack	OpenSec	39
	Three-Tier IDPS	97

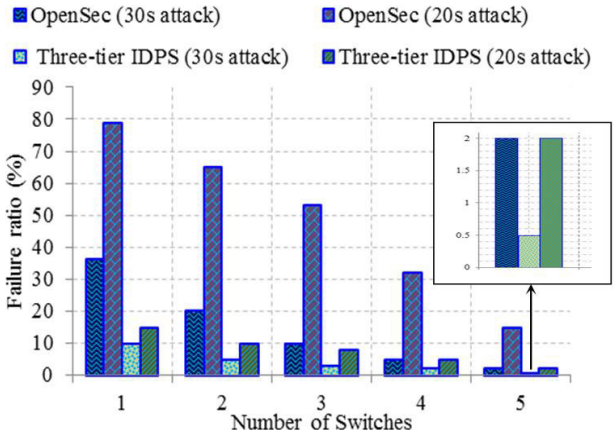

FIGURE 9. Comparison on Failure Rate.

OpenSec framework concentrates on detecting malicious packet in switches which fails to perform authentication and hence all the illegitimate users are allowed into the network. The Table 7 illustrates average detection rate for existing and proposed work, this average is given for all the switches. The detection ratios improved 38.72% at 25% and 58% at 50% of attacks. The proposed three-Tier IDPS also performs significantly better even when the attack intensity is higher.

The increase in detection ratio eventually decreases the failure ratio in our proposed system and increases in OpenSec framework. Fig. 9 demonstrates the achievement of failure rate with respect to number of switches. This results show three-Tier IDPS has lesser failure rate than OpenSec framework.

The failure rate decreases with the increase in switches; this is due to the sharing of packets that are received from users. The attack packets are launched at every 20 seconds and 30 seconds. The self-defined policies in data plane is not effective in predicting the malicious packets comparatively packet feature based prediction is effective. The packet features are the key to identify behavior of a user into the network. The decrease in failure ratio denotes the improvement of proposed system by using type-2 fuzzy system in OpenFlow switches. The decrease in failure rate and increase in detection rate ensures accurate detection of attack packets that mitigates excess consumption of resources in the network and hence the processing of normal packets is not slow downed. The restriction of malicious packets in OpenFlow switches enables to mitigate the attack packets in Tier 2 as well as Tier 3. Almost all the packets are detected in Tier 2 and only suspected packets and new packets are processed in Tier 3.

2) DELAY AND THROUGHPUT

Delay is a network metric that is required to be as low as possible. The lesser the delay will certainly improve network performance and enables faster processing of normal packets. In general, the increase in attack packets into the network will eventually increase delay and decrease throughput. The improvisation of these two metric in the presence of attack packets is challenging. In order achieve this; the provisioning of security in this network environment is strengthened. In this three-Tier IDPS the security is assisted with user validation, packet validation and flow validation by which the processing of normal packets are made faster. The involvement of attacks also reflects over these common network parameters and so comparison on delay and throughput is evaluated.

The growth in malicious packets into the network consumes larger resources and it leads to degrade performances for normal packets. The comparisons of delay and throughput parameters are also studied in Fig. 10 and Fig. 11 respectively. The mitigation of illegitimate users in three-Tier IDPS with the filtration of malicious packets enables to improve delay.

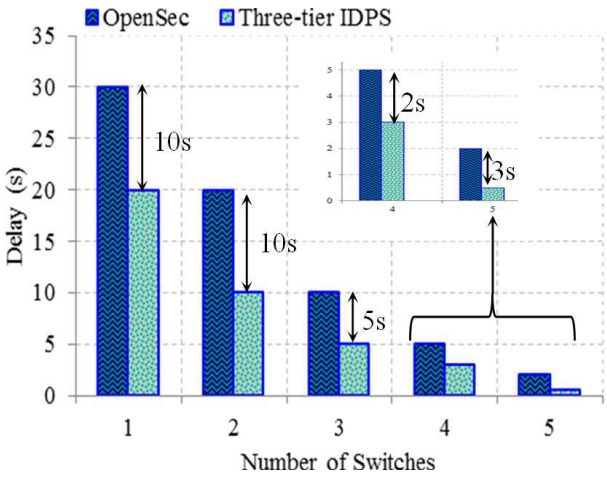

FIGURE 10. Comparison on Delay.

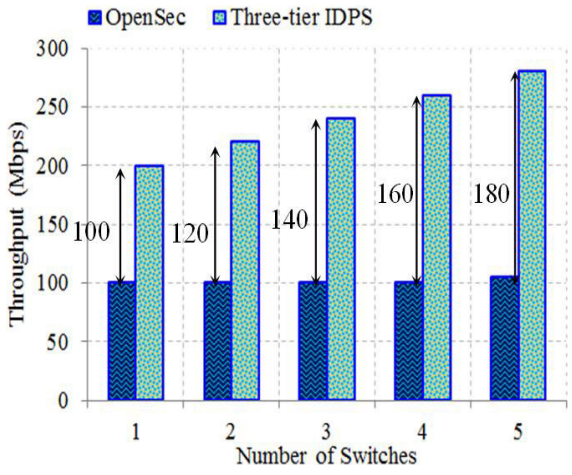

FIGURE 11. Comparison on Throughput.

Most of previous security works in SDN have majorly developed with the mitigation of malicious packets which also allows packets from illegitimate users. The delay decreases with the increase in switches, since the involvement of many switches will able to manage the flooded packets. Whereas in OpenSec framework all the users are allowed to submit packets which requires larger time for attack prediction that increases delay in processing normal packets.

The metric throughput increases while the bandwidth resource availability is sufficient for the packet to process. But, the bandwidth consumption is the major constraints which will be occupied when the attack packets increases. In three-Tier IDPS, the edge devices authenticate users and then the OpenFlow switches with type-2 filtration discards the malicious packets by analyzing the packet features. Almost all the malicious packets are discarded by the switches and even the suspicious flows are validated by controllers and the corresponding rule is updated into each switch for future prevention of those suspected packets. The prompt discarding of malicious packets increases the bandwidth availability and hence the throughput is increased when compared with OpenSec. The involvement of illegitimate users into the network is the major cause for increasing attack packets into the network.

From this measurement, the throughput is increased upto 50% from the previous OpenSec framework when compared with proposed three-Tier IDPS. The increase in throughput ensures that proposed three-Tier IDPS is accurate in prevention and detection of attacks in the network. The prevention of attack packets and attackers enables to mitigate new behaving attacks into the network.

3) TRAFFIC LOAD

Traffic load is the main parameter in IDPS where the participation of attackers is detected. Most commonly the attackers aim is to target a particular network entity and make it to drain all the resources. This is enabled by flooding the traffic into the network. The detection of malicious packets by the switches ensures to manage traffic load at switches. The appropriate detection of malicious packets will enable the management of traffic load and also provides prompt access for the normal packets based on their requested service. Traffic is the main cause of the attackers by which they enter into the network.

The comparison of traffic load for three-Tier IDPS and OpenSec is depicted in Figure 12 with respect to the increase in number of switches. The increase in abnormal traffic load in the network defines that there may be attack packets participation. The increase in traffic load at data plane tends to consume larger amount of bandwidth. This is reduced by allowing legitimate users into the network for requesting their service.

According to the increase in number of switches the traffic load is high in OpenSec than the proposed three-Tier IDPS. This denotes that arrived malicious packets are correctly identified in proposed three-Tier IDPS and hence the traffic

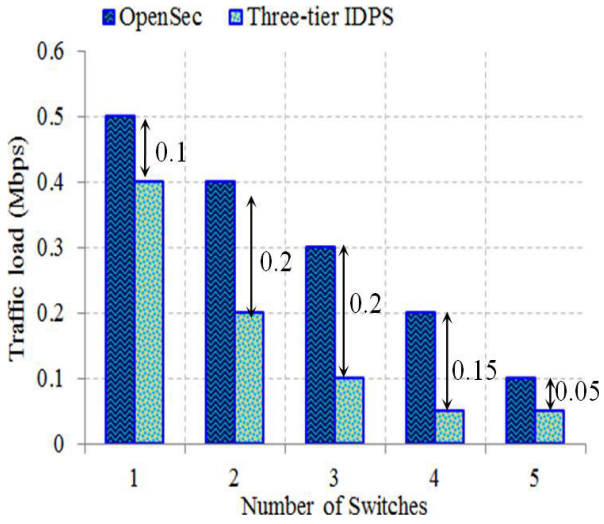

FIGURE 12. Comparison on Traffic Load.

load is comparatively less in our work. The increase in traffic load will gradually degrade the system performance and as a result the processing of normal packets will be reduced.

4) PRECISION AND ACCURACY

The detection of attacks is also performed by using popular machine learning algorithms for higher accuracy. These algorithms are employed for improving the processing speed and accuracy. Most of the works using these algorithms have been applied on control plane layer which is the major reason to trail this system for attack detection. The measurement of precision and accuracy is expressed as follows:

$$P = \frac{TP}{TP + FP} \tag{11}$$

$$AC = \frac{C_{MP}}{T_{MP}} \tag{12}$$

The term precision and accuracy is given as P and AC. The precision is computed from true positive and false positive that are represented as TP and FP. Here TP defines the correctly predicted attack packets from the arrived packets and FP defines the incorrectly predicted packets i.e. a normal packet as malicious. Further the accuracy is estimated from correctly detected malicious packets as C_{MP} with respect to the total number of malicious packets arrived that is denoted as T_{MP}. These two measures are significant while using in machine learning algorithm for attack detection. Machine learning algorithms are being the promising solution for making appropriate decision in attacks by analyzing the packet features and their behavior in the network. Hence, the machine learning based attack detection is compared.

The evaluated results of precision are depicted in Table 8 on which the results of machine learning algorithms are enlisted. The increase in the value of precision denotes the improvement in the detection of attacks in the network. The conventional machine learning algorithm is supposed to exists

TABLE 8. Precision of machine learning algorithms.

Mbps	SVM (%)	SOM (%)	SAE (%)	ANN (%)	Three-Tier IDPS (%)	
					Type-2 Fuzzy	Deep Learning
100	94.34	95.40	97.80	97	98	99
200	93.06	96.63	97.63	96.87	98	98.50
300	93.23	96.54	97.65	96.68	97	98.30

TABLE 9. Accuracy of machine learning algorithms.

Mbps	SVM (%)	SOM (%)	SAE (%)	ANN (%)	Three-Tier IDPS (%)	
					Type-2 Fuzzy	Deep Learning
100	94.56	96.85	97.98	98.30	98.50	99
200	94.23	96.67	97.67	98.12	98	99.20
300	94.12	96.78	96.90	97.85	99	99

with certain limitations as higher time consumption, kernel selection, parameter setting and others.

The accuracy in predicting malicious packets is illustrated in Table 9. As higher the accuracy it reflects in correct prediction of the participating attack packets. The use of type-2 fuzzy and deep learning algorithm increases precision as well as accuracy from which the attacks are detected and hence these methods are also suitable for real-world network environment.

VI. SECURITY ANALYSIS

Security is the key goal of this proposed three-Tier IDPS environment. This work focuses on both detection and prevention of intruders which is not widely focused in many of the previous research works. The process involved in each Tier is depicted in Table 10 from which the security provisioning in three-Tier IDPS is identified. The prevention of intruders is attained in two-fold as:

1) Firstly, authenticating users by RFID tags discards access to illegitimate users, but this way the unregistered intruders are reduced.
2) Secondly, the suspected packets are further analyzed at control plane then the corresponding new rules are installed into switches on type-2 fuzzy with which the new intruders are also prevented.

TABLE 10. Security provisioning in three-tier IDPS.

Tier	Layer	Method Used	Process
Tier 1	Edge Layer	User Validation	RFID with Signature
Tier 2	Data Plane Layer	Packet Validation	Packet Features Analysis
Tier 3	Control Plane Layer	Flow Validation	Port of the Flow is Verified

Hereby, the provisioning of security is also required to detect malicious packets accurately without any delay. Since, the delay in detection of attack packets leads to drain out resources and also extends processing time of the normal packets. So, the detection of attack is also followed in two-fold as:

1) Firstly, the malicious packets are detected in Tier 2 using type-2 fuzzy filtration based on the analysis of packet features.
2) Secondly, the suspected packets and new packets are processed in Tier 3 and identify the packet as malicious using deep learning method.

On behalf of detection and prevention of the intruders ensures to provide security in all aspects and the common requirements that are solved in security are detailed in the following. The common intention of the attackers is to steal the personal information for accessing the particular user's account and launch malicious packets.

(A) AUTHENTICATION – The authentication is attained by the use of RFID tags for individual users. The tag represents individual user and also a unique signature is generated to ensure the user is original. Hence authentication is achieved by allowing access only for the legal registered user tags.

(B) CONFIDENTIALITY – The confidentiality is obtained by maintaining the security credentials in secrecy. During authentication, the signatures and other credentials are not transmitted in raw form. It is encrypted before transmission and hence the confidentiality is assured.

(C) AVAILABILITY – This security requirement mainly targeted by DoS attack which is solved by monitoring network traffic and detecting the malicious behavior at data plane. The accurate prediction of attacks ensures availability in security.

(D) PRIVACY – The term privacy defines securing the private information of the user. According to this work, the private information is identity, tag identity and signature which are securely stored in the personal device and hence privacy is ensured in three-Tier IDPS.

(E) FORWARD SECURITY – In this three-Tier IDPS, the symmetric key is known only by the user and TA. Even the edge device has nil knowledge about the key. In this way, forward secrecy is obtained without leakage in security credential.

The use of RFID for authentication is also an effective solution in mitigating few attacks as:

(A) REPLAY ATTACK – In replay attack, the attacker will involve while the authentication response is delay. This attack is resisted in proposed three-Tier IDPS, since before verifying the security credential, the timestamp is validated. If he timestamp is invalid then the request from particular user will be certainly discarded.

(B) MIMA ATTACK – The MIMA is resisted during authentication, due to the submission of the security credentials in encrypted format or by using XOR operator. Due to this the propose three-Tier IDPS is not affected by MIMA.

(C) FORGERY ATTACK – The forgery attacks are involved when the tag of original user is used by another person. This attack is resisted in three-Tier IDPS since the user without knowing the identity is not able to finish authentication.

(D) DDOS ATTACK – The flooding of packets towards a particular target is known to be DDoS attack. The validation of packet features enables to restrict this type of attack into the system.

From this security analysis, our proposed three-Tier IDPS architecture is assured to provision security and also defends against attacks in the system. On comparison, the three-Tier IDPS is operated significantly better than the previous intrusion detection methods in SDN-IoT environment.

VII. CONCLUSION

In this paper, a three-Tier IDPS in SDN-IoT environment is designed especially to assure security. The provisioning of security is concentrated on Tier 1, Tier 2 and Tier 3. Firstly, in Tier 1 the Internet of Things (IoT) users are validated with RFID and signature that is encrypted using modified blowfish algorithm. Symmetric block cipher performs faster with stronger security and signature for each user is determined using SHA-256 algorithm. Secondly, in Tier 2 the OpenFlow switches are employed with type-2 fuzzy filtration that extracts packet features and detects malicious, normal and suspected packets. The suspected packets are then forwarded to tire 3 in which the flow of the packet is analyzed using deep learning method. In tire 3 the controllers maintain two queues as normal and suspicious. Even though two queues are only present they are not overloaded, since all the arrived packets are already processed in Tier 2 and filtered out. Therefore the packets are assisted with two queues in controllers. The processing of three-Tiers prevents intruders by authentication and new fuzzy rule installation. Similarly, the intruders are detecting by packet validation and flow validation process. On the whole, this proposed three-Tier IDPS ensures security in SDN-IoT environment by showing the better efficiency in terms of detection rate, failure rate, accuracy, precision, delay throughput and traffic load.

In future, we have planned to use block chain technology for authenticating users and implement the proposed system on large scale environment. Also this work can be extended with the selection of switches in order to reduce overloading of switches.

REFERENCES

[1] M. C. Dacier, H. König, R. Cwalinski, F. Kargl, and S. Dietrich, "Security challenges and opportunities of software-defined networking," *IEEE Secur. Privacy*, vol. 15, no. 2, pp. 96–100, Mar. 2017.

[2] S. Bera, S. Misra, and A. Jamalipour, "FlowStat: Adaptive flow-rule placement for per-flow statistics in SDN," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 3, pp. 530–539, Mar. 2019.

[3] A. Sallam, A. Refaey, and A. Shami, "On the security of SDN: A completed secure and scalable framework using the software-defined perimeter," *IEEE Access*, vol. 7, pp. 146577–146587, 2019.

[4] S. Zheng, "Research on SDN-based IoT security architecture model," in *Proc. IEEE 8th Joint Int. Inf. Technol. Artif. Intell. Conf. (ITAIC)*, Chongqing, China, May 2019, pp. 575–579.

[5] I. Farris, T. Taleb, Y. Khettab, and J. Song, "A survey on emerging SDN and NFV security mechanisms for IoT systems," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 812–837, 1st Quart., 2019.

[6] N. V. R. Kumar and M. Kumar, "Application of SDN for secure communication in IoT environment," *Comput. Commun.*, vol. 151, pp. 60–65, Feb. 2020.

[7] V. Varadharajan, K. Karmakar, U. Tupakula, and M. Hitchens, "A policy-based security architecture for software-defined networks," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 4, pp. 897–912, Apr. 2019.

[8] K. Sood, K. K. Karmakar, V. Varadharajan, U. Tupakula, and S. Yu, "Analysis of policy-based security management system in software-defined networks," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 612–615, Apr. 2019.

[9] Z. Yang and K. L. Yeung, "Flow monitoring scheme design in SDN," *Comput. Netw.*, vol. 167, Feb. 2020, Art. no. 107007.

[10] G. Huang and H. Y. Youn, "Proactive eviction of flow entry for SDN based on hidden Markov model," *Frontiers Comput. Sci.*, vol. 14, Jan. 2020, Art. no. 144502.

[11] N. Sultana, N. Chilamkurti, W. Peng, and R. Alhadad, "Survey on SDN based network intrusion detection system using machine learning approaches," *Peer-Peer Netw. Appl.*, vol. 12, no. 2, pp. 493–501, Mar. 2019.

[12] H. Peng, Z. Sun, X. Zhao, S. Tan, and Z. Sun, "A detection method for anomaly flow in software defined network," *IEEE Access*, vol. 6, pp. 27809–27817, 2018.

[13] M. Latah and L. Toker, "Towards an efficient anomaly-based intrusion detection for software-defined networks," *IET Netw.*, vol. 7, no. 6, pp. 453–459, Nov. 2018.

[14] Y. Cuo, F. Miao, L. Zhang, and Y. Wang, "CATH: An effective method for detecting denial-of-service attacks in software defined networks," *Sci. China Inf. Sci.*, vol. 62, Feb. 2019, Art. no. 32106.

[15] L. Fang, Y. Li, X. Yun, Z. Wen, S. Ji, W. Meng, Z. Cao, and M. Tanveer, "THP: A novel authentication scheme to prevent multiple attacks in SDN-based IoT network," *IEEE Internet Things J.*, early access, Sep. 27, 2019, doi: 10.1109/JIOT.2019.2944301.

[16] M. Wang, J. Liu, J. Mao, H. Cheng, J. Chen, and C. Qi, "RouteGuardian: Constructing secure routing paths in software-defined networking," *Tsinghua Sci. Technol.*, vol. 22, no. 4, pp. 400–412, Aug. 2017.

[17] C. Qi, J. Wu, H. Hu, and G. Cheng, "Dynamic-scheduling mechanism of controllers based on security policy in software-defined network," *Electron. Lett.*, vol. 52, no. 23, pp. 1918–1920, Nov. 2016.

[18] T. G. Nguyen, T. V. Phan, B. T. Nguyen, C. So-In, Z. A. Baig, and S. Sanguanpong, "SeArch: A collaborative and intelligent NIDS architecture for SDN-based cloud IoT networks," *IEEE Access*, vol. 7, pp. 107678–107694, 2019.

[19] T. V. Phan and M. Park, "Efficient distributed denial-of-service attack defense in SDN-based cloud," *IEEE Access*, vol. 7, pp. 18701–18714, 2019.

[20] A. Shenfield, D. Day, and A. Ayesh, "Intelligent intrusion detection systems using artificial neural networks," *ICT Express*, vol. 4, no. 2, pp. 95–99, Jun. 2018.

[21] S. K. Dey and M. M. Rahman, "Effects of machine learning approach in flow-based anomaly detection on software-defined networking," *Symmetry*, vol. 12, no. 1, p. 7, Dec. 2019.

[22] P. Manso, J. Moura, and C. Serrão, "SDN-based intrusion detection system for early detection and mitigation of DDoS attacks," *Information*, vol. 10, no. 3, p. 106, Mar. 2019.

[23] C. Birkinshaw, E. Rouka, and V. G. Vassilakis, "Implementing an intrusion detection and prevention system using software-defined networking: Defending against port-scanning and denial-of-service attacks," *J. Netw. Comput. Appl.*, vol. 136, pp. 71–85, Jun. 2019.

[24] M. Kim, "Supervised learning-based DDoS attacks detection: Tuning hyperparameters," *ETRI J.*, vol. 41, no. 5, pp. 560–573, Oct. 2019.

[25] M. A. Albahar, "Recurrent neural network model based on a new regularization technique for real-time intrusion detection in SDN environments," *Secur. Commun. Netw.*, vol. 2019, Nov. 2019, Art. no. 8939041.

[26] X. An, X. Zhou, X. Lü, F. Lin, and L. Yang, "Sample selected extreme learning machine based intrusion detection in fog computing and MEC," *Wireless Commun. Mobile Comput.*, vol. 2018, Jan. 2018, Art. no. 7472095.

[27] B. S. Khater, A. W. B. A. Wahab, M. Y. I. B. Idris, M. A. Hussain, and A. A. Ibrahim, "A lightweight perceptron-based intrusion detection system for fog computing," *Appl. Sci.*, vol. 9, no. 1, p. 178, Jan. 2019.

[28] D. Arivudainambi, K. A. V. Kumar, and S. S. Chakkaravarthy, "LION IDS: A meta-heuristics approach to detect DDoS attacks against software-defined networks," *Neural Comput. Appl.*, vol. 31, no. 5, pp. 1491–1501, May 2019.

[29] Y. Xiao, Z.-J. Fan, A. Nayak, and C.-X. Tan, "Discovery method for distributed denial-of-service attack behavior in SDNs using a feature-pattern graph model," *Frontiers Inf. Technol. Electron. Eng.*, vol. 20, no. 9, pp. 1195–1208, Sep. 2019.

[30] A. Mansour, M. Azab, M. R. M. Rizk, and M. Abdelazim, "Biologically-inspired SDN-based intrusion detection and prevention mechanism for heterogeneous IoT networks," in *Proc. IEEE 9th Annu. Inf. Technol., Electron. Mobile Commun. Conf. (IEMCON)*, Vancouver, BC, Canada, Nov. 2018, pp. 1120–1125.

[31] I. H. Abdulqadder, D. Zou, I. T. Aziz, B. Yuan, and W. Li, "SecSDN-cloud: Defeating vulnerable attacks through secure software-defined networks," *IEEE Access*, vol. 6, pp. 8292–8301, 2018.

[32] A. M. Zarca, D. Garcia-Carrillo, J. B. Bernabe, J. Ortiz, R. Marin-Perez, and A. Skarmeta, "Enabling virtual AAA management in SDN-based IoT networks," *Sensors*, vol. 19, no. 2, p. 295, Jan. 2019, doi: 10.3390/s19020295.

[33] M. Allouzi and J. Khan, "SafeFlow: Authentication protocol for software defined networks," in *Proc. IEEE 12th Int. Conf. Semantic Comput. (ICSC)*, Laguna Hills, CA, USA, Jan. 2018, pp. 374–376.

[34] N. Abbas, M. Asim, N. Tariq, T. Baker, and S. Abbas, "A mechanism for securing IoT-enabled applications at the fog layer," *J. Sensor Actuator Netw.*, vol. 8, no. 1, p. 16, Feb. 2019.

[35] S. Kaur and M. Singh, "Hybrid intrusion detection and signature generation using deep recurrent neural networks," *Neural Comput. Appl.*, vol. 32, no. 12, pp. 7859–7877, Jun. 2020.

[36] M. Ambrosin, M. Conti, F. De Gaspari, and R. Poovendran, "LineSwitch: Tackling control plane saturation attacks in software-defined networking," *IEEE/ACM Trans. Netw.*, vol. 25, no. 2, pp. 1206–1219, Apr. 2017.

[37] T. Ha, S. Yoon, A. C. Risdianto, J. Kim, and H. Lim, "Suspicious flow forwarding for multiple intrusion detection systems on software-defined networks," *IEEE Netw.*, vol. 30, no. 6, pp. 22–27, Nov. 2016.

[38] R. Wang, Z. Jia, and L. Ju, "An entropy-based distributed DDoS detection mechanism in software-defined networking," in *Proc. IEEE Trustcom*, Aug. 2015, pp. 310–317.

[39] A. Lara and B. Ramamurthy, "OpenSec: Policy-based security using software-defined networking," *IEEE Trans. Netw. Service Manage.*, vol. 13, no. 1, pp. 30–42, Mar. 2016.

[40] Q. Yan, Q. Gong, and F. R. Yu, "Effective software-defined networking controller scheduling method to mitigate DDoS attacks," *Electron. Lett.*, vol. 53, no. 7, pp. 469–471, Mar. 2017.

[41] C.-T. Li, C.-Y. Weng, and C.-C. Lee, "A secure RFID tag authentication protocol with privacy preserving in telecare medicine information system," *J. Med. Syst.*, vol. 39, no. 8, p. 77, Aug. 2015.

[42] M.-S. Hwang and C.-C. Lee, "Research issues and challenges for multiple digital signatures," *Int. J. Netw. Secur.* vol. 1, pp. 1–7, Jul. 2005.

[43] C.-T. Chen and C.-C. Lee, "A two-factor authentication scheme with anonymity for multi-server environments," *Secur. Commun. Netw.*, vol. 8, no. 8, pp. 1608–1625, May 2015.

[44] H. Min-Shiang, L. Cheng-Chi, and T. Yuan-Liang, "Two simple batch verifying multiple digital signatures," in *Information and Communications Security*, vol. 2229, S. Qing, T. Okamoto, and J. Zhou, Eds. Berlin, Germany: Springer, 2001, pp. 233–237.

[45] J. M. Mendel and D. Wu, "Critique of 'a new look at type-2 fuzzy sets and type-2 fuzzy logic systems,'" *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 3, pp. 725–727, Jun. 2017.

[46] H.-K. Lim, J.-B. Kim, J.-S. Heo, K. Kim, Y.-G. Hong, and Y.-H. Han, "Packet-based network traffic classification using deep learning," in *Proc. Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC)*, Okinawa, Japan, Feb. 2019, pp. 46–51.

AMIR ALI received the M.Sc. degree in information technology and the M.Phil. degree in computer science. He is currently pursuing the Ph.D. degree in computer science with the PUCIT, University of the Punjab, Lahore, Pakistan. He is also serving as a Lecturer with the Department of Statistics and Computer Science, UVAS, Lahore. He has also done Internationally recognized Microsoft and Cisco Certifications, such as MCP, MCSA, MCSE, CCNA, CCNP (Routing and Switching), and CCVP (VOIP). He has over ten years of teaching experience. His research interests include security issues in software defined networking and computer networks.

MUHAMMAD MURTAZA YOUSAF received the Ph.D. degree from the University of Innsbruck, Austria, in 2008. He worked on networks for grid computing during his Ph.D. He is currently an Assistant Professor with the PUCIT, University of the Punjab, Lahore, Pakistan. He is an Active Researcher and a Supervising Postgraduate Student at M.S. and Ph.D. levels. He has authored and coauthored journal and conference papers at national and international level in the field of computer science. His current areas of research include computer networks, parallel and distributed computing, cloud computing, and data science and interdisciplinary research.