Computers, Materials & Continua
DOI:10.32604/cmc.2021.015568
Article

Tech Science Press

Intelligent Framework for Secure Transportation Systems Using Software-Defined-Internet of Vehicles

Mohana Priya Pitchai[1], Manikandan Ramachandran[1,*], Fadi Al-Turjman[2] and Leonardo Mostarda[3]

[1]School of Computing, SASTRA Deemed University, Thanjavur, 613 401, India
[2]Research Centre for A.I. and IoT, Near East University, Nicosia, Mersin, 10, Turkey
[3]Department of Software Engineering, Camareno University, Italy
[*]Corresponding Author: Manikandan Ramachandran. Email: manikandan75@core.sastra.edu

Received: 28 November 2020; Accepted: 29 January 2021

Abstract: The Internet of Things plays a predominant role in automating all real-time applications. One such application is the Internet of Vehicles which monitors the roadside traffic for automating traffic rules. As vehicles are connected to the internet through wireless communication technologies, the Internet of Vehicles network infrastructure is susceptible to flooding attacks. Reconfiguring the network infrastructure is difficult as network customization is not possible. As Software Defined Network provide a flexible programming environment for network customization, detecting flooding attacks on the Internet of Vehicles is integrated on top of it. The basic methodology used is crypto-fuzzy rules, in which cryptographic standard is incorporated in the traditional fuzzy rules. In this research work, an intelligent framework for secure transportation is proposed with the basic ideas of security attacks on the Internet of Vehicles integrated with software-defined networking. The intelligent framework is proposed to apply for the smart city application. The proposed cognitive framework is integrated with traditional fuzzy, crypto-fuzzy and Restricted Boltzmann Machine algorithm to detect malicious traffic flows in Software-Defined-Internet of Vehicles. It is inferred from the result interpretations that an intelligent framework for secure transportation system achieves better attack detection accuracy with less delay and also prevents buffer overflow attacks. The proposed intelligent framework for secure transportation system is not compared with existing methods; instead, it is tested with crypto and machine learning algorithms.

Keywords: Internet of things; smart cities; software-defined network; intelligent transportation system; fuzzy inference system

1 Introduction

Vehicular Adhoc Network (VANET) is a highly dynamic network [1]. VANET is a three-layered network architecture where the bottom layer is composed of vehicles. The middle layer is composed of Road Side Units (RSU), and the top layer has Central Authority (C.A.) which acts as a brain for the VANET. It makes decisions on how the routing of vehicles should be done.

As VANET is highly dynamic, the network architecture is susceptible to malicious attacks [2]. One of the most common threats found is flooding attacks. These can be originated among the three layers such as flooding attacks among the vehicles, flooding attacks between vehicles and RSU's and flooding attacks between RSU's and C.A.

Nowadays, the Internet of Things (IoT) [3] plays a predominant role in controlling all the devices from the central component that automates secure routing based on the network situation. In this research work, traditional VANET architecture is considered Internet of Vehicles (IoV) [4], where all the vehicles status can be obtained and monitored via many technologies through wireless communication. Those technologies vary in protocols and include the well-known technologies such as Wi-Fi IEEE 802.11p [5], WAVE IEEE 1609 [6], WiMAX IEEE 802.16 [7], Bluetooth Low Energy (BLE) [8] and Zigbee [9]. These protocols are enabled in the vehicles participating in the road traffic. They help obtain the vehicle I.D., range, timestamp, and Received Signal Strength (RSS) between the vehicle and the RSU's.

IoV is mostly susceptible to one such threat named Sybil attack [10], which is originated as flooding attacks in two different ways including vehicle claiming with fake I.D. to obtain the certificate from the authority and vehicle claiming with multiple fake I.D.s. Sybil attacks are categorized as vehicle-to-vehicle (among the vehicles) and vehicle-to-network-infrastructure (from the vehicle to other layers of the VANET). In case of flooding attacks, reconfiguring an entire network infrastructure is challenging, and network customization is also not feasible in VANET and IoV. This research work focuses on establishing a flexible IoV environment by designing a cognitive framework for secure transportation while utilizing Software Defined Networks (SDN) [11,12].

SDN is also the three-layered network paradigm in which the bottom plane has an Open Flow (OF) switches to collect the data from the participating vehicles in the street's traffic. In SDN, the vehicles are enabled with wireless communication technologies to share its I.D., timestamp, and location through Global Positioning System (GPS) [13]. The central plane is the control plane which consists of a controller [14], and it acts as a brain for the SDN while the top plane is an application plane in which user-designed applications are incorporated.

In a traditional VANET, all vehicles are connected to the OF switch [15–17], which acts as a medium for request and response message communications. Additionally, the OF switch is connected to both the RSU and the SDN controller. The SDN provides a global view of the network topology based on the road situations used to make fair decisions by the controller to avoid network congestion. SD-IoV [18] is also susceptible to various security compromises and mostly for Sybil compromise. The adverse effects of Sybil attack include data tampering, fails in fair resource allocation and data aggregation.

Smart Cities [19] are developed with automated technologies that have internet-connected devices integrated into them. Those devices are composed of IoT infrastructural components like sensors for data collection and actuators to send request-response to and from the network. Smart Cities are composed of many real-time applications like smart agriculture [20], smart home system [21], smart healthcare system [22], and smart transportation [23]. The conceptual idea of Smart Cities is to control the internet-connected devices from a centralized controller. This research paper focuses on smart and secure transportation in SD-IoV, which can detect common security attacks of VANET and mitigates those attacks by deploying suitable defence mechanisms.

The proposed cognitive framework for secure transportation comprises three modules including traffic generation using sumo simulator, feature extraction and attack mitigation. The cognitive framework detects security attack scenarios and makes the framework aware of those situations.

Traffic congestions are then reduced when compared to the existing methods. The proposed cognitive framework is deployed using fuzzy rules for feature extraction, and the attack detection is based on the defined fuzzy rules set. The proposed framework fairly allocates resources for the participating vehicles, authenticates the incoming request packets from the vehicles sends suitable responses to the components available in the network infrastructure.

The proposed intelligent framework applies to the industrial transportation sectors like road and rail, airlines, airfreight and logistics, trucking, airport services, highways, and rail tracks to congestions on the road traffic flows.

The proposed research article's objective is to survey the existing methods for attack detection in IoV, understand the feasibility of deployed algorithms at each phase of SD-IoV, and detect and mitigate security attacks in SD-IoV and to compare with the available methods. The non-flexible IoV environment motivates to incorporate with SDN to customize the network based on the situations and road conditions and alert the vehicles about secure routing.

The proposed cognitive framework's contributions include detecting security attacks in the SDN controller deployed in SD-IoV architecture using a crypto fuzzy rule set. The proposed cognitive framework mitigates those security attacks' adverse effects by providing fair resource allocation, authenticating internet-connected devices, and preventing data tampering as SDN applications. Data communications among SD-IoV infrastructure is done by a fuzzy crypto designer (Incorporated with fuzzy crypto rules) rather than a traditional fuzzy logic designer. This research article is organized in the following order:

Section 1 provides an introduction into VANET, IoV and SDN, SD-IoV, and Smart Cities. It also provides the problem statement, objectives and contributions of the work. Section 2 presents a literature review of the existing works that have been done on detecting attack traffic flows in SD-IoV. Section 3 explains the proposed cognitive framework; Section 4 presents the experimental network setup. Section 5 provides a detailed analysis of the results of the experiments conducted. Section 6 summarises critical takeaways from this research work and provides recommendations for future research directions.

2 Literature Review

Dibaei et al. [24] discussed the major security attacks on intelligent connected vehicles. It is the only research work which classifies defence methods into four categories (cryptography, network security, software vulnerability detection and malware detection). Only a few security attacks are considered. Tahsien et al. [25] discussed machine learning-based security solutions for IoT systems, attacks on various phases of SDN. The research work concentrated on the possible attacks in IoT rather than considering DDoS attacks alone. The research work is domain-dependent which is deployed with only machine learning algorithms to detect possible security attacks. Zhang and Lu [26] discussed the communication performance of the vehicle self-organize network. The research work aims at the urban road traffic scenarios of IoT-based Intelligent Transportation System. Have the advantages of low research and development risk and using mature technology. Implementation cost is relatively high since the technology used is for self-organizing vehicle network. Wen Chen et al. [27] discussed tracing the source and used a statistics-based traceback scheme using SDN architecture. Consumes a few network resources Time consumption to trace the source is lower compared to other existing methods. Traceback scheme is attacked dependent as it only detects DDoS attack traffic flows in Software-Defined Network-based Smart Cities. False Positive Rate (FPR) is high on attack detection accuracy.

Sherazi et al. [28] discussed the Intrusion Prevention System (IPS) for DDoS attack on the Internet of Vehicles (IoV) environment. Artificial Intelligence (A.I.) and Machine Learning (ML) based approaches such as fuzzy logic, and Q-learning based approach is integrated with IDS to detect DDoS attack. A helpful tool for improving security in the next-generation complex heterogeneous networking against the sophisticated attacks for implementing the sustainable vehicular network. The proposed method is attacked dependent, as it is only suitable for DDoS attack data types. Galeano-Brajones et al. [29] discussed the stateful SDN security solutions for IoT traffic that detects and mitigates Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks based on the concept of entropy as the detection method. Entropy-based DDoS attack detection method has high sensitivity, low False Positive Rate (FPR) for adjustment of algorithm parameters. No need to deploy additional network device with the detection method. Open Flow (OF) switches are limited to install and comply with the rules for forwarding stateless packets received from the network control plane, which generates signalling overhead and increase in latency. Liu et al. [30] discussed the game model to analyze the attack benefit between attacker and defender. Proposed an enhanced Distributed Low-Rate Attack Mitigating (eDLAM) mechanism which maintains the lightweight Malicious Request Tables (MRT). The proposed eDLAM mechanism performs better in terms of False Negative Rate (FNR) reduced by 10.5% and False Positive Rate (FPR) reduced by 44% compared with existing mechanisms. Has less latency on detecting and recovering from the attacks. Sharma et al. [31] discussed the various security aspects of the Internet of Vehicles (IoV), including security requirements, challenges and attacks. The authors have proposed a lightweight authentication protocol for Radio Frequency Identification Devices (RFID). Performance of the proposed lightweight authentication protocol for Radio Frequency Identification Devices (RFID) has better performance in low detection time, low CPU and memory consumption to strengthen the existing IoV environment. It is not suitable to use in scenarios where the memory of Open Flow (OF) switches is less for storing the OF tables and not having enough time to perform the key setup. Arif et al. [32] discussed security and privacy issues in VANET's. The research also addresses the effectiveness of VANETs and cloud computing with related security and privacy issues. The authors had just surveyed about the security and privacy issues in VANETs, and the work is not implemented to detect security attacks using intelligence algorithms. Security issues in the VANETs are inherent and distinctive. It is difficult to predict such attacks in VANETs because of network size, high mobility, repeated topological changes, and various applications and services. Singh et al. [33] discussed machine learning approaches used like gradient boosting classifier, Neural net, Logistic regression, Decision tree, Naive Bayes, Linear Support Vector Machine, K-Nearest Neighbor and Random Forest algorithms. To test a centralized SDN controller's performance in detecting DDoS attack traffic flows in Vehicle-to-Infrastructure (V2I) communication. Gradient boost classifier achieves better attack detection accuracy with less false alarm rates. The proposed attack detection mechanism works finely even with the encrypted payload as it relies on unsupervised data training. The detection mechanism also runs in parallel with the North Bound Applications without an overhead. The attack detection mechanism is suitable only for a centralized SDN controller. Not suitable for other SDN planes. It is not suitable for high scalable road traffic environment based on vehicular density. Jabbar et al. [34] discussed the impact of Packet-In flooding attacks in the SDN controller. The author focused on analyzing the impact of Packet-In flooding based DDoS attacks on SD-IoV controllers. This work is an initiative of SD-IoV security which motivates to have deeper insights into DDoS attacks on this network infrastructure. Reduced overall network throughput. Huge increase in a load of SDN controller. The research work analyzed the impact of a centralized SDN controller, whereas multiple controllers are not considered.

3 Proposed Cognitive Framework

The proposed cognitive framework comprises five phases: data collection, data storage, data processing, data transmission, and data delivery. In phase1, data is collected from the sensors which are integrated on the vehicles through GPS. In phase 2, the collected data is stored in two different ways. The first way is done via offline data, and the second way is achieved by capturing the online traffic data from both the vehicles and RSU's. In phase-3, data processing occurs where fuzzy rules are used for feature extraction followed by a Restricted Boltzmann Machine (RBM) algorithm to detect attack traffic flows. This specific algorithm is selected as its stochastic supports this dynamic network topology. RBM algorithm is suitable for raw traffic flows while the crypto fuzzy logic designer is used to encrypt packet traffic flows.

Additionally, in this phase, applicable mitigation procedures are applied to detect malicious traffic flows. In phase 4, data transmission takes place where the processed data gets sent to the requested receiver. In phase-5, data delivery to the recipient is ensured without delay in time or alteration of data. The data alteration includes sequence modification, content modification and time modification. The different phases of the proposed cognitive framework are shown below in Fig. 1.

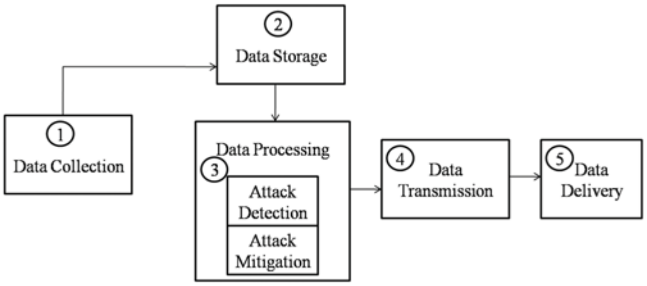


Figure 1: Phases of the proposed cognitive framework

3.1 *Data Collection*

Data collection phase begins with a traffic generation module which contains online, and offline traffic flow samples from the OF enabled switches. All OFPT requests and OFPT response messages from the associated sensors with the vehicles crossing the road signals are collected and stored for future analysis. The request and response messages are collected from vehicles participating in the roadside traffic on the data plane and RSU components.

3.2 *Data Storage*

Data storage phase deals with the SDN controller to store the network traffic flow samples for future analysis. Data storage also depends on the number of Open Flow rules installed from the control plane and the data plane switches. For the newly incoming OFPT requests, the SDN controller starts looking up for the OF rules incorporated within the OF switches. If a rule exists regarding the OF switch entry, the controller proceeds with that rule. Otherwise, those network traffic flow OFPT requests and responses messages are then forwarded to the controller for decision making.

3.3 Data Processing

Data processing applies intelligent algorithms to the collected data. The processed data can then be used for various purposes, such as securing the network. The collected data is processed, and the network can be secured by detecting attack traffic flows. After detecting these attacks, suitable mitigating procedures can be implemented to mitigate those attacks. In this research work, data processing consists of two phases. The first phase attacks are detected by finding Distributed Denial of Service (DDoS) and the Sybil traffic flows attacks. In the second phase, attack mitigation procedures are implemented. Online traffic flows of SD-IoV architecture is analyzed from the collected data. When data simulation was carried out, it was evident that the vehicles created network congestion. It is due to sending multiple request packets to the RSU to claim location-based certificate from multiple forged I.D.'s. It was detected using both traditional and crypto fuzzy rules. It consists of assigned OFPT request messages from the data plane switches, and response messages from the control plane controllers. Data processing starts with feature extraction for the sake of dimensionality reduction. Four features are extracted from the collected dataset, including Vehicle_ID, Vehicle Step_time, Vehicle_type and Vehicle_speed out of ten features in total to detect traffic flows attacks. This process presented in a schematic diagram in Fig. 2.

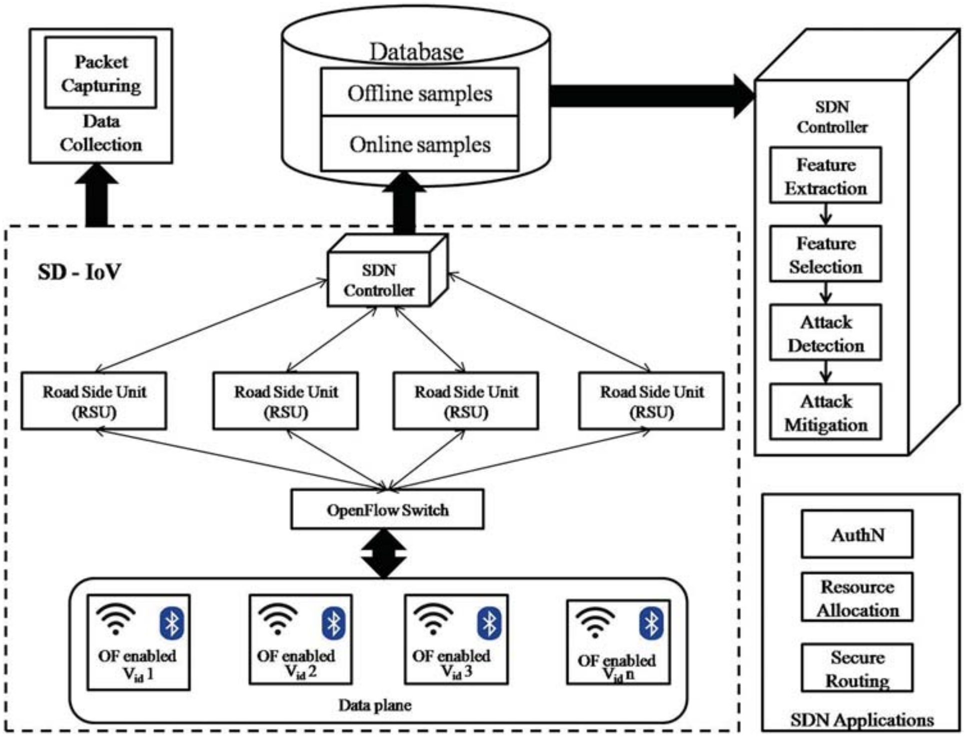


Figure 2: Proposed intelligent framework

Algorithms 1 and 2 present the algorithms used in the attack detection process using traditional fuzzy rules and fuzzy crypto rules. DDoS and Sybil attack detection in SD-IoV using traditional fuzzy is applied to the attributes collected in the dataset. Algorithm 1 represents the pseudocode for the attack detection process. The module declares input variables and output variables to the fuzzy logic designer, followed by defining all variables' membership functions. Fuzzy rules can be created for various membership functions using conditional statements. Fuzzy rules can incorporate boolean conditions such as "and", "or" or a combination of "and & or" to define rules for attack detection. The same logic is applied in the pseudocode presented in Algorithm 2, where fuzzy crypto rules are used.

Algorithm 1: Pseudocode for attack detection using traditional fuzzy rules

```
Input: Feature Vector F = {Vehicle_ID, Vehicle Step_time, Vehicle_type, Vehicle_speed, No. of. OFPT request messages}
Output: Attack traffic flows
    1: Choose Fuzzy Logic Designer (FLD) as mamdani
    2: Define Input Variables (IV)
IV={Vehicle_ID, Vehicle Step_time, Vehicle_type, Vehicle_speed, No. of. OFPT request messages}
    3: Define Membership Functions (M.F.) for IV
        • Each Input Variable is assigned with values in MF
        • Vehicle_ID as a numerical integer value
        • Vehicle Step_time in milliseconds
        • Vehicle_type as bus, car
        • Vehicle_speed as flow location
        • No. of. OFPT request messages as a numeric integer value
    4: Define Output Variable (O.V.)
    OV = {Sybil Attack, DDoS attack traffic flows}
    5: Create Fuzzy Rules in Fuzzy Rule Base by mapping each input variable with its membership function (Rule Editor with verbose and symbolic representations)
    (e.g.,): if Vehicle_ID is 3 and Vehicle Step_time ranges between 3 to 3.444 ms and Vehicle_type as bus & car and Vehicle_speed as East to North and No. of. OFPT request messages are >50
    if Vehicle_ID is 3 &Vehicle Step_time ranges between 3 to 3.444 ms & Vehicle_type as bus, car & Vehicle_speed as East to North & No. of. OFPT request messages are >50
```

Algorithm 2: Pseudocode for attack detection using crypto fuzzy rule

```
Input: Feature Vector F = {Vehicle_ID, Vehicle Step_time, Vehicle_type, Vehicle_speed, No. of. OFPT request messages}
Output: Attack traffic flows
    1: Choose Fuzzy Logic Designer (FLD) as mamdani
    2: Define Input Variables (IV)
        IV = {Vehicle_ID, Vehicle Step_time, Vehicle_type, Vehicle_speed, No. of. OFPT request messages}
    3: Define Membership Functions (MF) for IV
```

(Continued)

- Each Input Variable is assigned with values in MF
- Vehicle_ID as numerical integer value
- Vehicle Step_time in milliseconds
- Vehicle_type as bus, car
- Vehicle_speed as flow location
- No. of. OFPT request messages as numeric integer value

4: Define Output Variable (O.V.)

OV = {Sybil Attack, DDoS attack traffic flows}

5: Create Fuzzy Rules in Fuzzy Rule Base by mapping each input variable with its membership function (Rule Editor with verbose and symbolic representations)

(e.g.,): if sign(M, PKV^{-1}) is 3 and if Vehicle Step_time (M, PKV^{-1}) ranges between 3 to 3.444 ms and if Vehicle_type(M, PKV^{-1}) as bus & car and if Vehicle_speed(M, PKV^{-1}) as East to North and if No.of. OFPT request messages is from (M, PKV^{-1}) > 50

if sign(M, PKV^{-1}) is 3 & if Vehicle Step_time (M, PKV^{-1}) ranges between 3 to 3.444 ms & if Vehicle_type (M, PKV^{-1}) as bus, car & if Vehicle_speed (M, PKV^{-1}) as East to North & if No. of. OFPT request messages is from (M, PKV^{-1}) > 50

Sybil and DDoS attack in SD-IoV can also be detected using Neural Network's RBM algorithm by finding the energy consumption of OF switches deployed to send and receive OFPT request and response messages. Attack detection using the RBM algorithm is computed using energy consumption parameters based on malicious nodes and selfish nodes. The energy configuration of RBM algorithm [35] is given by,

$$E(V,H) = \sum_{i=1}^{m}\sum_{j=1}^{n} w_{m,n}v_m h_n - \sum_{i=1}^{m} a_{im}v_m - \sum_{j=1}^{n} b_{jn}h_n \tag{1}$$

where,

$v_m =$ Visible layer of RBM

$h_n =$ Hidden layer of RBM

$w_{m,n}v_m h_n =$ Weight matrix of a visible and hidden layer of RBM

a_{im}, $b_{jn} =$ Bias values

The visible input layer of RBM is fed with five input values as a feature vector represented as Input Feature Vector (IFV). The hidden layer processes the input values given in the visible layer.

The probability distribution function of the RBM is given by,

$$P(V,H) = \frac{1}{Z}E(V,H) \tag{2}$$

where,

Z = constant value which partitions visible and hidden layer.

RBM consists of two different gradients, including positive and negative. The positive gradient is used to train the network with input features, whereas the negative gradient phase is used to test the network with more sample data. Here, RBM is created with a single visible layer of five input features including Vehicle_ID, Vehicle_Steptime, Vehicle_Speed, Vehicle_type and No. of.

request packets from a vehicle_ID. The hidden layer then learns the feature values, and in case of newly incoming OFPT request traffic flows, the hidden layer (output layer) maps the values with all the features and detects malicious traffic flows. The attack detection output is represented with a malicious vehicle I.D. RBM component is deployed as a separate module in the SDN controller, which can be used for various real-time applications. Malicious attacks are detected based on the OF switches' energy consumption based on their active and idle cases. Traffic flows attacks are also detected based on the OF rules for the internet-connected vehicles. Selfish Nodes (S.N.) represents the idle nodes not participating in data transmission of the network environment. Energy consumed by S.N. is computed as follows,

$$SN = \frac{1}{Z} \exp\{-E(V, H)\}[REM_{energy}] \tag{3}$$

where,

Z = constant value which partitions visible and hidden layer.

$[REM_{energy}]$ = Remaining energy of selfish OF switches

Malicious attacks [36] are detected by implementing several OF rules for the participating vehicles, represented using Eq. (4).

$$MN = \frac{1}{Z} \exp\{-E(V, H)\}\left[REM_{energy}\right][OFPT_{Reqpackets}] \tag{4}$$

where,

Z = constant value which partitions visible and hidden layer.

$[REM_{energy}]$ = Remaining energy of selfish of switches

$\left[OFPT_{Reqpackets}\right]$ = Open Flow Protocol Request Packets

$E(V, H) = EnergyConfigurationofRBMalgorithm$ deployed in the network

Attack mitigation procedures are initiated with Authentication (AuthN) module from the SDN application plane in case of malicious OFPT requests originated from a forged Vehicle_ID or multiple Vehicle_ID's. AuthN module also checks for a location-based certificate issued by the RSU. Vehicles with different signal strength for RSU would not be admitted in the final verification. As North Bound–Application Programming Interface (NB-API) acts as an interface between SDN application and control plane; routing module gathers information about the road conditions. It provides routes that are vehicle free to avoid network congestions and delay in reaching the destination. By invoking AuthN module, vehicles with multiple I.D.'s are denied at the entry-level. Hence, the network bandwidth and processing time of OFPT requests significantly lower than the existing methods.

3.4 Data Transmission

The processed OFPT request and response are transmitted over dynamic sensors. To and from the vehicles and the RSU components from the SDN controller through the OF switches.

3.5 Data Delivery

The proposed cognitive framework also ensure the data is transmitted to the receiver without modification in parameters such as time, content, and sequence.

Fig. 3. shows the workflow of the proposed cognitive framework in which the framework begins with packet capturing from OF-enabled switches in the data plane. Traffic flows are then feature extracted to detect DDoS or Sybil attacks. Fig. 3. represents a single fuzzy rule for one suspected vehicle in which attack detection begins with verifying four features with suspected attributes including V_type, V_ID, V_Steptime and V_Speed. The vehicle with I.D. starting with the value 3 with a step time ranging between 3.1 and 3.444, and has a vehicle type of bus or car travelling in the EasttoNorth direction causes a network congestions on the travelled path. Suppose the same vehicle with ID-3 is requesting a security certificate from different RSU, the RSU checks for the vehicle's signal strength. In the case of valid OFPT request, it is forwarded to the Central Authority for issuing a certificate. The invalid request is forwarded to the adjacent RSU to check the vehicle signal strength. In case of malicious traffic flows with forged vehicle_ID and tampered data, SDN controller automates suitable mitigation procedures from the SDN application plane.

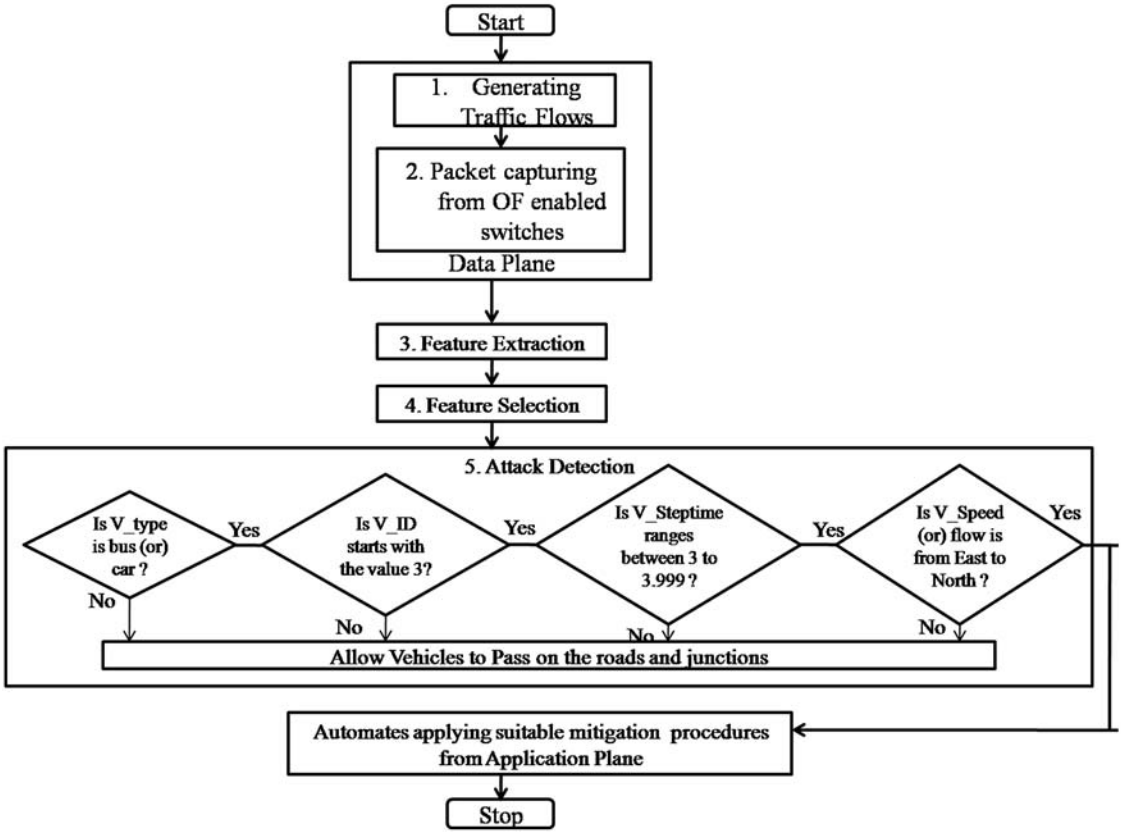

Start
1. Generating Traffic Flows
2. Packet capturing from OF enabled switches
Data Plane
3. Feature Extraction
4. Feature Selection
5. Attack Detection
Is V_type is bus (or) car ?
Yes
No
Is V_ID starts with the value 3?
Yes
No
Is V_Steptime ranges between 3 to 3.999 ?
Yes
No
Is V_Speed (or) flow is from East to North ?
Yes
No
Allow Vehicles to Pass on the roads and junctions
Automates applying suitable mitigation procedures from Application Plane
Stop

Figure 3: Flow chart of the proposed cognitive framework

The analysis of the proposed intelligent framework's complexity is found that it is applicable for the small-scale networks. This research work is tested only with four switches and applied only for the road traffic transportation sector. Since the proposed framework is not tested with various transportation sectors, it is not found about attack detection accuracy in the rest of the transportation sectors. The proposed framework works fine in small scale networks, and as a future research direction, validation is done for the large-scale road network traffic scenario.

The attack detection procedure is done only with the base vehicle metrics such as Vehicle_ID, Vehicle Step_time, Vehicle_type, Vehicle_speed, No. of. OFPT request messages, not all vehicle metrics are considered. The road traffic data set is captured only for a specific region named Europarc Roundabout, Creteil, France.

4 Experimental Network Setup

In this research work, the dataset is collected by generating traffic flows by designing SD-IoV network topology using Mininet emulator and Simulation of Urban MObility (SUMO)to simulate the VANET. SD-IoV architecture is simulated for both legitimate network traffic flows and Sybil attack situations with different scenarios. Further analysis was performed on the dataset published by the Microscopic Vehicular mobility trace of Europarc roundabout, Creteil, France. The dataset's trace contains details on the vehicles specifically about buses passed in the Europarc Creteil roundabout between 17:00 AM to 19:00 AM. The dataset also contains timestep, vehicle_slope, vehicle_lane, vehicle_angle, vehicle_type, vehicle_position, vehicle_speed with vehicle_ID information.

In Mininet emulator, the network topology is designed with 4 switches, each with two hosts. Its gateway I.P. address is 172.2.0.1 and server I.P. address is 10.1.0.1 for Network Address Translator (NAT) and router I.P. address is assigned as 172.2.0.2 with gateway I.P. address as 10.2.0.1. Fig. 4. provides an illustration of the simulation, the traces shows that hosts connected to switch 1 (S1) is comprising of IP addresses for h11 (172.2.0.11) and h12 (172.2.0.12), in switch 2 (S2) is comprising of IP addresses for h21 (172.2.0.21) and h22 (0.0.0.0), in switch 3 (S3) is comprising of IP addresses for h31 (172.2.0.31) and h32 (0.0.0.0), in switch 4 (S4) is comprising of IP addresses for h41 (172.2.0.41) and h42 (172.2.0.42). Two added hosts with I.P. addresses h13 as 172.2.0.12 is connected to s1 and h23 as 172.2.0.12 is connected to s2.

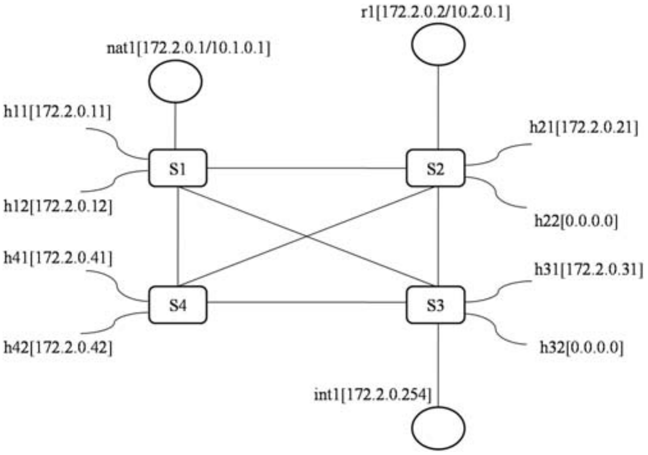
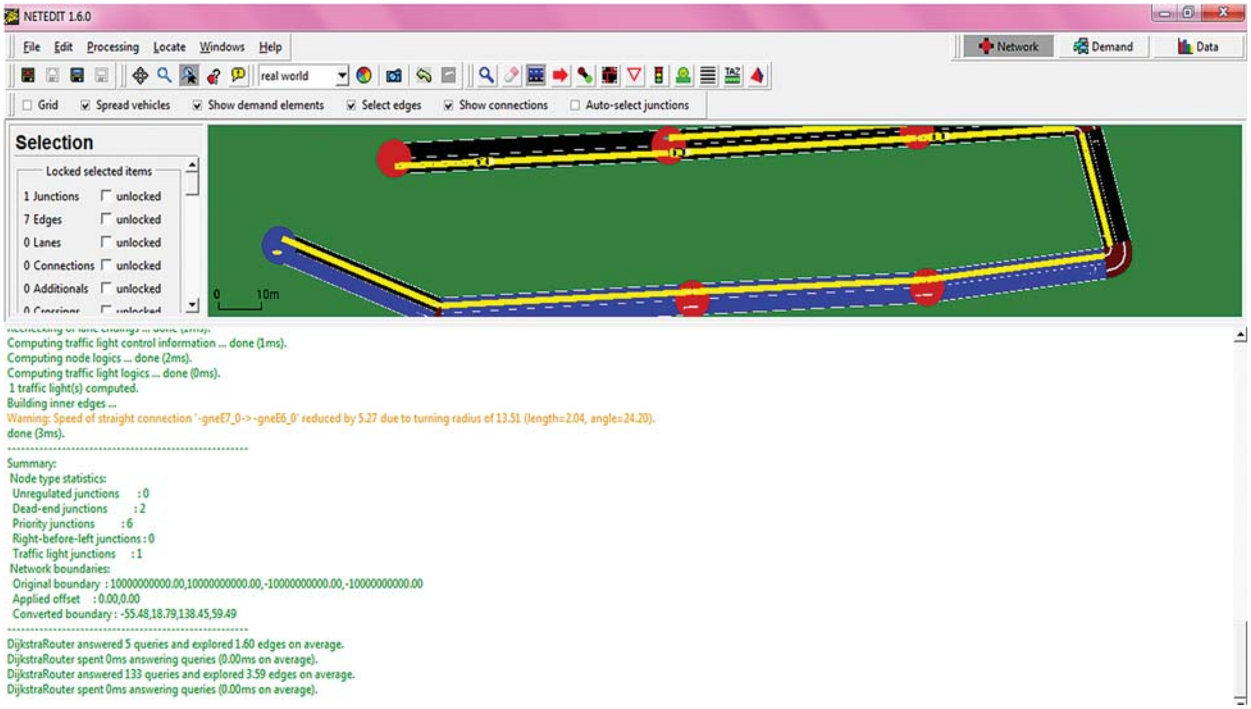


Figure 4: Mininet simulation diagram

Fig. 5. shows the VANET topology created using Simulation of Urban Mobility (SUMO) tool in which 9 junctions are created with 18 edges, and 6 routes. 4 of the routes are used by vehicles. In contrast, the rest of the 2 routes are used by individuals. Two-way roads have also been created with Crosswalks and simulated. The traffic light is placed at Junction 1. Details about the simulated topology can be found in Tab. 1.



☐ Grid ☑ Spread vehicles ☑ Show demand elements ☑ Select edges ☑ Show connections ☐ Auto-select junctions

1 Junctions	☐ unlocked
7 Edges	☐ unlocked
0 Lanes	☐ unlocked
0 Connections	☐ unlocked
0 Additionals	☐ unlocked
0 Crossings	☐ unlocked

```
Computing traffic light control information ... done (1ms).
Computing node logics ... done (2ms).
Computing traffic light logics ... done (0ms).
 1 traffic light(s) computed.
Building inner edges ...
Warning: Speed of straight connection '-gneE7_0->-gneE6_0' reduced by 5.27 due to turning radius of 13.51 (length=2.04, angle=24.20).
done (3ms).
-----------------------------------------------------
Summary:
 Node type statistics:
  Unregulated junctions      : 0
  Dead-end junctions         : 2
  Priority junctions         : 6
  Right-before-left junctions : 0
  Traffic light junctions    : 1
 Network boundaries:
  Original boundary  : 10000000000.00,10000000000.00,-10000000000.00,-10000000000.00
  Applied offset     : 0.00,0.00
  Converted boundary : -55.48,18.79,138.45,59.49
-----------------------------------------------------
DijkstraRouter answered 5 queries and explored 1.60 edges on average.
DijkstraRouter spent 0ms answering queries (0.00ms on average).
DijkstraRouter answered 133 queries and explored 3.59 edges on average.
DijkstraRouter spent 0ms answering queries (0.00ms on average).
```

Figure 5: SUMO simulation diagram

Table 1: Simulation of VANET

Roadjunctions	Roadedges	Routes	Vehicles	Persons
J1	J1-J2, J2-J1	(R1) J1-J2-J3	V1 V10	Person_0
J2	J2-J3, J3-J2	(R2) J1-J2	V2 V11	Person_1
J3	J3-J4, J4-J3	(R3) J1-J2-J3-J4-J5	V3 V12	Person_2
J4	J4-J5, J5-J4	(R4) J4-J5-J6-J7-J8	V4 V13	
J5	J5-J6, J6-J5	(R5) J6-J7-J8	V5 V14	
J6	J6-J7, J7-J6	(R6) J6-J7-J8-J9	V6 V15	
			V7 V16	
			V8 V17	
			V9 V18	

Tab. 1 shows the simulation details of the created VANET topology. It consists of attributes such as Junctions, Edges, Routes, Vehicles and Individuals. Vehicle ID 3 is considered a malicious node that tries to use the location certificate issued by RSU. Multiple requests from the same vehicle ID 3 originated from obtaining a certificate from RSU, resulting in flooding-based DDoS attack. Each RSU component is connected to the Python-based SDN controller named POX, which regulates the road traffic flows and avoids network congestions by placing OF rules on the vehicles' participating vehicles in the road traffic flows. For attack (or) malicious traffic flows detection, the generated data is trained using MATLAB tool's fuzzy inference system discussed next to Tab. 1.

The captured network traffic flows are fed into the feature extraction module for extracting the attributes specific values from the raw network traffic data. From the extracted features of IoV, feature selection logic is applied to detect the security attacks occurred, and the same is applied for online network traffic flows. Security attack traffic flows are detected based on the fuzzy rule set defined in the mamdani fuzzy logic designer. The combination of attributes in the collected dataset is fed as a specific membership function specified with ranges. The fuzzy rules are then created for each target labels. Fig. 6. shows a schematic representation of the input variables into the fuzzy logic designer, which then maps out a target output label. Fig. 7. represents how input variables may result in a Sybil attack where various Fuzzy Inference Systems (FIS) models are plotted for the inputs and the targeted outputs in Fig. 8. Both traditional and designer Crypto fuzzy logic designer are incorporated with public-key cryptosystem. It is for message exchanges among the SD-IoV infrastructural components. It adds rules for the input attributes. Crypto fuzzy logic designer handles encrypted request-response messages of SD-IoV in OF_REQ_PACKETS and OF_RES_PACKETS format.

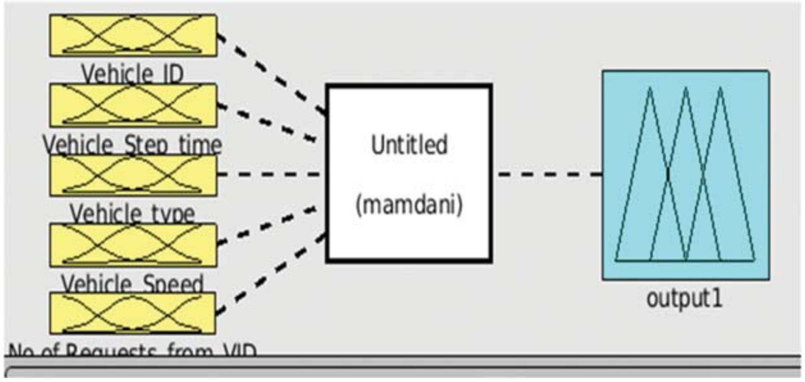


Figure 6: Defining input variables

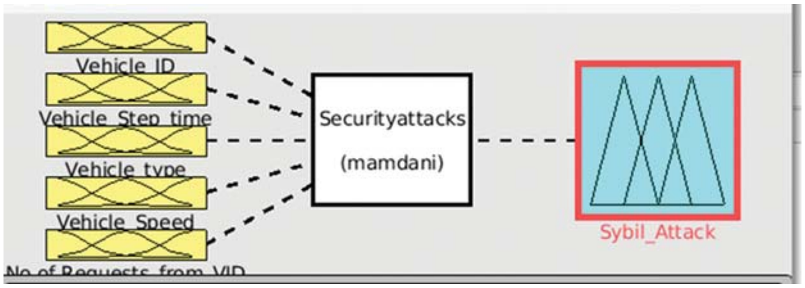


Figure 7: Defining input variables for sybil attack detection

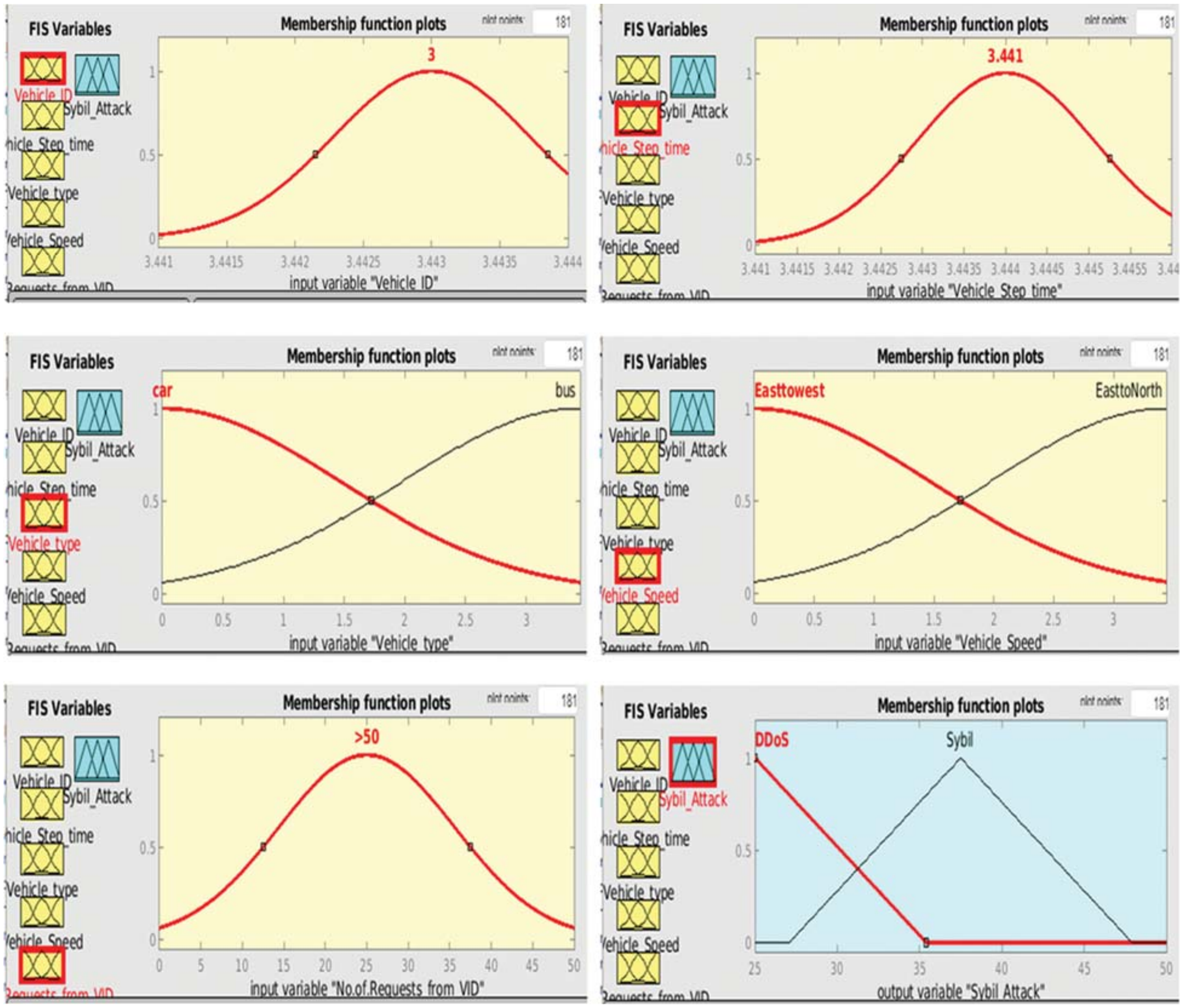

FIS Variables
Membership function plots
plot points: 181
Vehicle ID
Sybil_Attack
hicle Step time
Vehicle type
Vehicle Speed
Requests from VID
3
3.441
input variable "Vehicle ID"
input variable "Vehicle Step time"
car
bus
input variable "Vehicle type"
Easttowest
EasttoNorth
input variable "Vehicle Speed"
>50
input variable "No.of.Requests from VID"
DDoS
Sybil
output variable "Sybil Attack"

Figure 8: Defining membership functions for input and output variables

5 Results and Discussions

This section discusses the results obtained for parameters such as AuthN of incoming OF requests, Sybil attack detection, Performance of Sybil attack detection accuracy by traditional fuzzy and proposed crypto-fuzzy inference system, delay in the processing of Open Flow Protocol (OFPT) request messages, resource allocation and buffer overflow attacks. This research work was tested with the parameters mentioned above using traditional fuzzy, crypto-fuzzy and RBM deployed in SD-IoV, and the work was then compared with the IoV. The microscopic dataset is collected for 1 h and 2 min.

Figs. 9. and 10 shows the traffic analysis of incoming vehicles on the road between the 9 simulated junctions. The observed traffic flows of negative patterns were stored on the SDN controller to detect the new pattern of malicious traffic flows with correlated features. Fig. 9 illustrates a high increase in the number of request packets when the time frame is precisely 3.114 min. It is observed in the simulator that a lot of OFPT request packets from vehicle ID 3 is originated for a location-based security certificate. Fig. 9. is illustrated concerning vehicle step_time and number of OFPT request packets sent to the controller whereas Fig. 10. is illustrated concerning vehicle_ID and number of OFPT request packets sent to the controller.

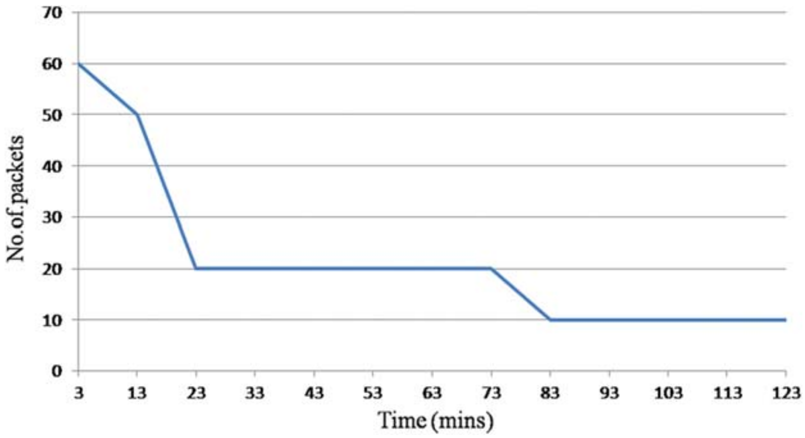


Figure 9: Incoming traffic analysis of road junctions

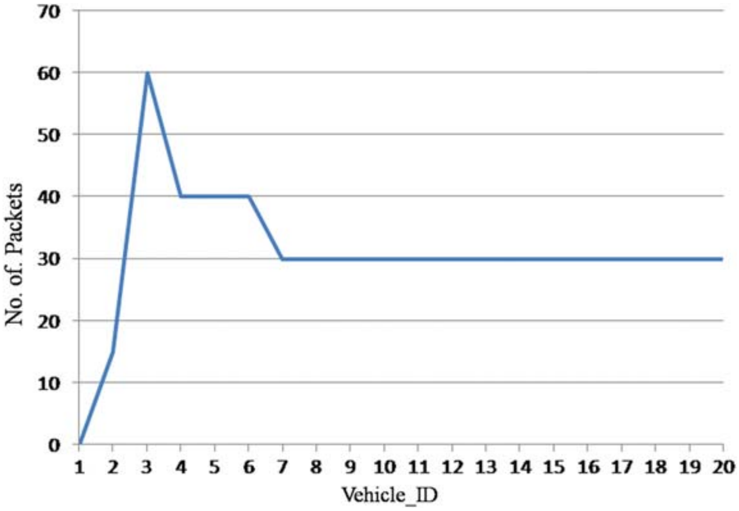


Figure 10: Incoming traffic analysis of road junctions with vehicle_ID

For this research, three defence methods, including traditional fuzzy, crypto-fuzzy, and RBM, have been used to detect DDoS attacks and Sybil attacks. Fig. 11 illustrates the detection accuracy for each of those methods. Traditional fuzzy and crypto-fuzzy detects attacks only for the vehicle_ID's trained data, whereas the vehicle_ID's whereas RBM detection method detects Sybil attack traffic flows in an unsupervised fashion. It is reflected in Fig. 11 where RBM based detection shows the highest accuracy of detection. RBM based detection method correlates the values of incoming traffic flows of vehicles. It was observed in the simulator that the negative pattern of flooding OFPT request was due to vehicle_ID15. Traditional fuzzy and crypto fuzzy detection methods are only suitable for supervised data. It was evident since they only detected vehicle_ID 3 but not vehicle_ID15.

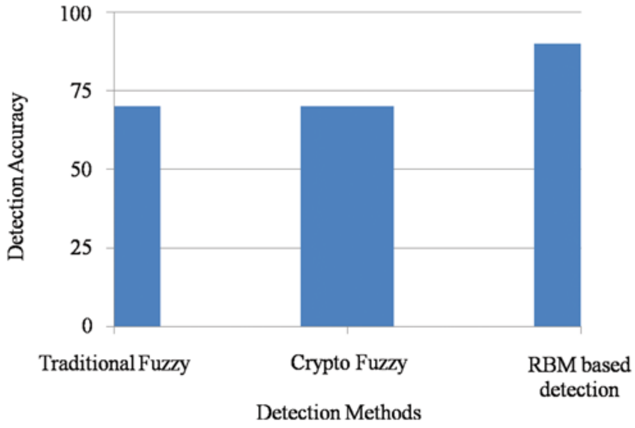


Figure 11: Sybil attack detection accuracy

Fig. 12 illustrates the delay in processing OFPT request packets, in which vehicle_ID 3 is connected to third RSU among the junctions, resulting in massive flood request packets to the controller. Vehicle_ID is spoofed as random source I.P. addresses to flood the packets to the controller, which results in the high delay of processing legitimate traffic flows. Among the nine connected junctions, the higher delay is observed in RSU_ID 3 from SDN controller as it is dumped with attack traffic flows by denying legitimate OFPT request traffic flows.

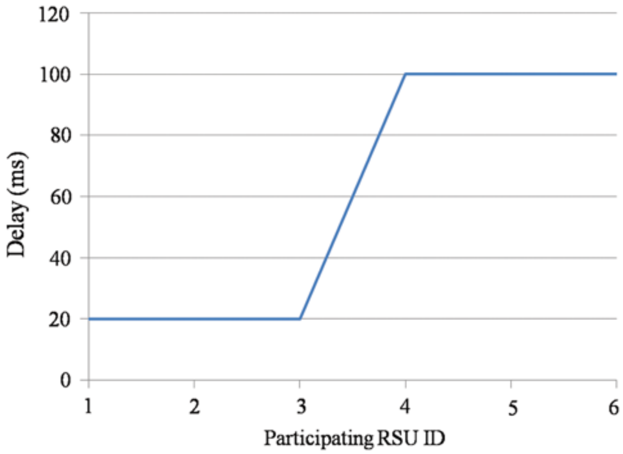


Figure 12: Traffic analysis on delay in the processing of OFPT request traffic flows

Fig. 13 provides a representation of the delay metric when the SDN controller is deployed with various three defending mechanisms. These mechanisms are deployed in the participating RSU's. The delay in processing all OF the request traffic flow is slightly reduced for RSU ID 1, 2, 3 in fuzzy approaches. In contrast, the RBM based detection method significantly reduced the delay in all the RSU's.

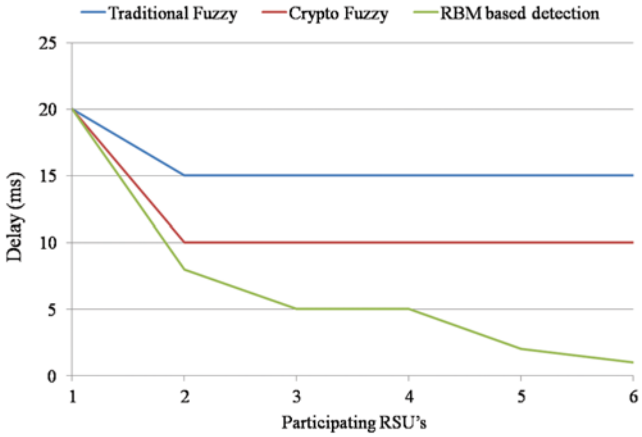


Figure 13: Delay in the processing of OFPT request traffic flows

Here, the delay metric is calculated based on the number of legitimate incoming OFPT request from the switches and the successfully processed OFPT response traffic flows from the controller-RSU-vehicles.

Fig. 14 illustrates the impact of buffer overflow attacks on the OF switches during the SDN controller rules implementation. Buffer overflow attacks are also measured based on the OFPT request packets reaching the OF switches and RSU's. Traffic analysis was done for both cases and Fig. 15 shows how buffer overflow attacks are reduced once the defence mechanisms are deployed on the OF switch. SDN controller does not process all OF the request packets; malicious traffic flows are denied and not processed further, which results in rapid decreasing of OF rule implementation on the vehicles.

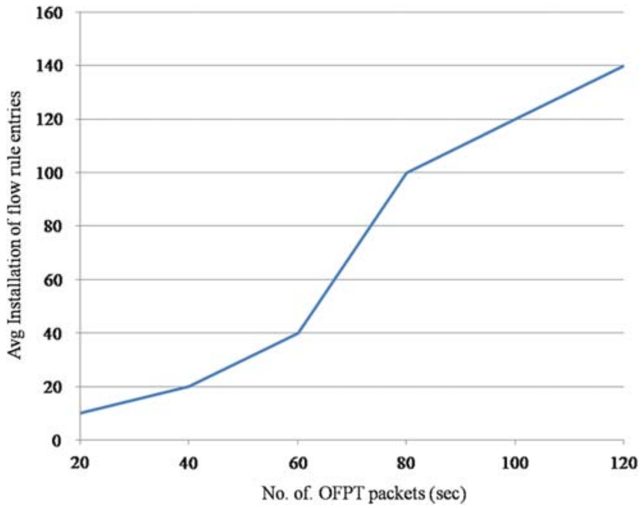


Figure 14: Installation of flow rule entries

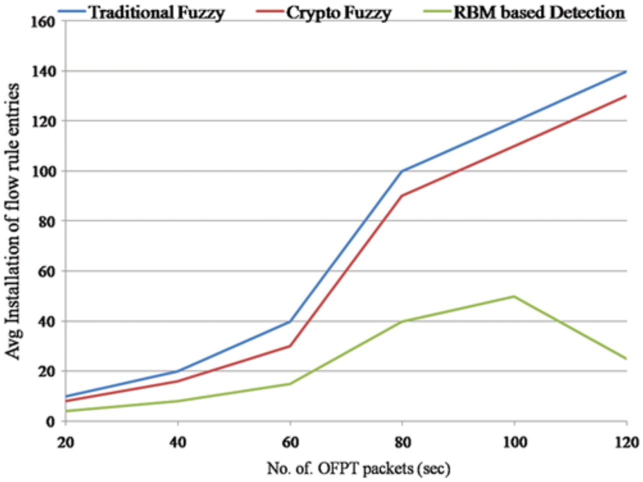


Figure 15: Installation of flow rules during buffer overflow attacks

6 Conclusion

This research paper proposed an intelligent framework to provide a secure transportation system for smart cities. The framework was integrated using the traditional fuzzy, crypto-fuzzy and RBM algorithms to detect attack traffic flows in a Software-Defined–Internet of Vehicles network infrastructure. The framework was also further developed to prevent buffer overflow attacks in OF switches and it provides less delay about one third of the time taken is found in processing the OFPT requests. Among the three algorithms used, it is found that RBM based DDoS attack detection achieves higher accuracy with unsupervised learning capability. For future research, it is recommended to deploy the scenario in real-time and compare the results with existing methods to test performance using the SDN controller's various metrics.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] H. Ahmed, S. Pierre and A. Quintero, "A flexible testbed architecture for vehicular adhoc network," *Vehicular Communications*, vol. 9, pp. 115–126, 2017.

[2] M. Arif, G. Wang, M. Z. A. Bhuiyan, T. Wang and J. A. Chen, "A survey on security attacks in vehicular adhoc networks: Communication, applications and challenges," *Vehicular Communications*, vol. 19, no. 2, pp. 1–36, 2019.

[3] P. Goyal, A. K. Sahoo and T. K. Sharma, "Internet of things: Architecture and enabling technologies," *Materials Today Proceedings*, vol. 34, no. 3, pp. 719–735, 2020.

[4] S. Sharma and B. Kaushik, "A survey on internet of vehicles: Applications, security issues & solutions," *Vehicular Communications*, vol. 20, no. 4, pp. 100182, 2019.

[5] C. M. Huang, M. S. Chiang, D. T. Dao, H. M. Pai, S. Xu *et al.,* "Vehicle-to-infrastructure (V2I) offloading from cellular network to 802.11p Wi-Fi network based on the software-defined network architecture," *Vehicular Communications*, vol. 9, no. 1, pp. 288–300, 2017.

[6] R. Ramanathan, "An empirical study on MAC layer in IEEE 802.11p/WAVE based vehicular adhoc networks," *Procedia Computer Science*, vol. 143, no. 7, pp. 720–727, 2018.

[7] A. Bacioccola, C. Cicconetti, C. Eklund, L. Lenzini, Z. Li *et al.,* "IEEE 802.16: History, status and future trends," *Computer Communications*, vol. 33, no. 2, pp. 113–123, 2010.

[8] W. B. Jaballah, M. Conti and C. Lal, "Security and design requirements for software-defined vehicular ad-hoc networks," *Computer Networks*, vol. 169, pp. 107099, 2020.

[9] G. S. Chirayil and A. Thomas, "A study on cost effectiveness and security of vehicular ad-hoc network technologies for future enhancement," *Procedia Technology*, vol. 25, pp. 356–363, 2016.

[10] B. Yu, C. Z. Xu and B. Xiao, "Detecting sybil attacks in vehicular ad-hoc networks," *Journal of Parallel and Distributed Computing*, vol. 73, no. 6, pp. 746–756, 2013.

[11] M. Jammal, T. Singh and A. Shami, "Software defined networking: State of art and research challenges," *Computer Networks*, vol. 72, no. 4, pp. 74–98, 2014.

[12] B. A. A. Nunes, M. Mendonca and X. Nguyen, "A survey of software-defined networking: Past, present and future of programmable networks," *IEEE Communication Surveys and Tutorial*, vol. 16, no. 3, pp. 1617–1634, 2014.

[13] M. A. Hossain, I. Elshafiey and A. Al-Sanie, "Cooperative vehicular positioning with vehicular ad-hoc network in urban environments," in *IEEE Asia-Pacific Conf. on Applied Electromagnetics*, Langkawi, Malaysia, pp. 393–396, 2016.

[14] Y. Zhao, L. Iannone and M. Riguidel, "On the performance of software defined network controllers: A reality check," in *IEEE Conf. on Network Function Virtualization and Software Defined Network*, San Francisco, CA, USA, pp. 79–85, 2015.

[15] H. Kim and N. Feamster, "Improving network management with software defined networking," *IEEE Communications Magazine*, vol. 51, no. 2, pp. 114–119, 2013.

[16] F. Al-Turjman and C. Altrjman, "Enhanced medium access for traffic management in smart-cities, vehicular-cloud," *IEEE Intelligent Transportation Systems Magazine*, Early Access, 2020.

[17] F. Al-Turjman and S. Alturjman, "5G/IoT-enabled UAVs for multimedia delivery in industry-oriented applications," *Multimedia Tools and Applications*, vol. 79, no. 13–14, pp. 1–22, 2018.

[18] K. Indira, P. Ajitha, V. Reshma and A. Tamizhselvi, "An efficient secured routing protocol for software defined internet of vehicles," in *Int. Conf. on Computational Intelligence in Data Science*, Chennai, India, pp. 1–4, 2019.

[19] S. K. Singh, Y. S. Jeong and J. H. Park, "A deep learning-based IoT-oriented infrastructure for secure smart city," *Sustainable Cities and Society*, vol. 60, pp. 102252, 2020.

[20] F. Bu and X. Wang, "A smart agriculture IoT system based on deep reinforcement learning," *Future Generation Computer Systems*, vol. 99, no. 6, pp. 500–507, 2019.

[21] D. Mocrii, Y. Chen and P. Musilek, "IoT-based smart homes: A review of system architecture, software, communications, privacy and security," *Internet of Things*, vol. 1, no. 6, pp. 81–98, 2018.

[22] S. Baskar, P. Mohamed Shakeel, R. Kumar, M. A. Burhanuddin and R. Sampath, "A dynamic and interoperable communication framework for controlling the operations of wearable sensors in smart healthcare applications," *Computer Communications*, vol. 149, no. 8, pp. 17–26, 2020.

[23] J. Lin, J. Niu, H. Li and M. Atiquzzaman, "A secure and efficient location-based service scheme for smart transportation," *Future Generation Computer Systems*, vol. 92, no. 2, pp. 694–704, 2019.

[24] M. Dibaei, X. Zheng, K. Jiang, R. Abbas, S. Liu *et al.,* "Attacks and defences on intelligent connected vehicles: A survey," *Digital Communications and Networks*, vol. 6, no. 4, pp. 399–421, 2020.

[25] S. M. Tahsien, H. Karimipour and P. Spachos, "Machine learning based solutions for security of Internet of Things (IoT): A survey," *Journal of Network and Computer Applications*, vol. 161, pp. 102630, 2020.

[26] H. Zhang and X. Lu, "Vehicle communication network in intelligent transportation system based on internet of things," *Computer Communications*, vol. 160, pp. 799–806, 2020.

[27] W. Chen, S. Xiao, L. Liu, X. Jiang and Z. Tang, "A DDoS attacks traceback scheme for SDN-based smart city," *Computers & Electrical Engineering*, vol. 81, pp. 106503, 2020.

[28] H. H. R. Sherazi, R. Iqbal, F. Ahmad, Z. A. Khan and M. H. Chaudary, "DDoS attack detection: A key enabler for sustainable communication in internet of vehicles," *Sustainable Computing: Informatics and Systems*, vol. 23, pp. 13–20, 2019.

[29] J. Galeano-Brajones, J. Carmona-Murillo, J. F. Valenzuela-Valdés and F. Luna-Valero, "Detection and mitigation of DOS and DDOS attacks in iot-based stateful sdn: An experimental approach," *Sensors*, vol. 20, no. 3, pp. 816, 2020.

[30] G. Liu, W. Quan, N. Cheng, H. Zhang and S. Yu, "Efficient DDOS attacks mitigation for stateful forwarding in Internet of Things," *Journal of Network and Computer Applications*, vol. 130, no. 3, pp. 1–13, 2019.

[31] S. Sharma and B. Kaushik, "A survey on internet of vehicles: Applications, security issues & solutions," *Vehicular Communications*, vol. 20, no. 4, pp. 100182, 2019.

[32] M. Arif, G. Wang, M. Z. A. Bhuiyan, T. Wang and J. Chen, "A survey on security attacks in vehicular adhoc networks: Communication, applications and challenges," *Vehicular Communications*, vol. 19, no. 2, pp. 100179, 2019.

[33] P. K. Singh, S. K. Jha, S. K. Nandi and S. Nandi, "ML-based approach to detect ddos attack in vehicle to infrastructure communication under sdn architecture," in *10th Proc. of TENCON Conf.*, Jeju, Korea (South), pp. 144–149, 2018.

[34] A. J. Siddiqui and A. Boukerche, "On the impact of DDOS attacks on software-defined internet-of-vehicles control plane," in *Proc. of Int. Wireless Communications & Mobile Computing Conf.*, Limassol, Cyprus, pp. 1284–1289, 2018.

[35] P. Mohana Priya and K. R. Manjula, "Cog-SDN: Mitigation mechanism for distributed denial of service attacks in software defined networks," in *Int. Conf. on Applications and Techniques in Information Security*, Thanjavur, India, Springer, pp. 202–215, 2019.

[36] P. MohanaPriya and S. M. Shalinie, "Restricted boltzmann machine-based cognitive protocol for secure routing in software defined wireless networks," *IET Networks*, vol. 6, no. 6, pp. 162–168, 2017.