electronics

MDPI

Article

Secure Intelligent Vehicular Network Using Fog Computing

Samuel Kofi Erskine and Khaled M. Elleithy *

Computer Science and Engineering Department, University of Bridgeport, Bridgeport, CT 06604, USA; serskine@my.bridgeport.edu
* Correspondence: elleithy@bridgeport.edu; Tel.: +1-203-576-4703

Received: 16 February 2019; Accepted: 18 April 2019; Published: 24 April 2019

Abstract: VANET (vehicular ad hoc network) has a main objective to improve driver safety and traffic efficiency. The intermittent exchange of real-time safety message delivery in VANET has become an urgent concern due to DoS (denial of service) and smart and normal intrusions (SNI) attacks. The intermittent communication of VANET generates huge amount of data which requires typical storage and intelligence infrastructure. Fog computing (FC) plays an important role in storage, computation, and communication needs. In this research, fog computing (FC) integrates with hybrid optimization algorithms (OAs) including the Cuckoo search algorithm (CSA), firefly algorithm (FA), firefly neural network, and the key distribution establishment (KDE) for authenticating both the network level and the node level against all attacks for trustworthiness in VANET. The proposed scheme is termed "Secure Intelligent Vehicular Network using fog computing" (SIVNFC). A feedforward back propagation neural network (FFBP-NN), also termed the firefly neural, is used as a classifier to distinguish between the attacking vehicles and genuine vehicles. The SIVNFC scheme is compared with the Cuckoo, the FA, and the firefly neural network to evaluate the quality of services (QoS) parameters such as jitter and throughput.

Keywords: VANET; fog computing; DoS attacks and intrusion; FFBPNN; Cuckoo search algorithm; firefly algorithm; QoS parameters

1. Introduction

It is noticeable that the automation industry has substantially improved in the last couple of years. The integration of hardware and software components produces better drivability and customer satisfaction. A vehicular ad hoc network (VANET) contains mobile vehicles with on-board processing units (OBPU) and roadside units (RSUs) that assist vehicles [1–3]. Vehicle-to-vehicle (V2V) communication is fortified to provide improved information to the drivers regarding roadside accidents, traffic jams, etc. This improves driver safety and the driving comfort of the vehicle in city traffic and on highways [4]. Highways, crossroads conditions, weather conditions, and vehicles monitoring are now part of the VANET important safety applications that must be complied. Examples of the safety applications include: Slow stop vehicle advisor (SSVA), post-crash notifications (PCN), and collision/congestion avoidance (CCA). These safety applications are important for VANET. VANET utilizes these safety applications to acquire prior knowledge of crossroads, highways, and knowledge of other vehicles conditions. In addition, safety applications enable drivers to execute sound judgment. Through safety applications, drivers are capable to obtain real-time information needed in order to enable them to initiate logical judgment and prevent further road and highway accidents occurrence. Regarding SSVA, vehicles that have slowed down or halted convey messages or information while utilizing warning signals message received from the network and take appropriate action. The warning signal messages sensitizes the surrounding vehicles in the VANET that may be in danger. With regard to PCN, messages are conveyed to highway patrols for further assistance through neighboring vehicles. Neighboring vehicles are closer to each other such that trust establishment in them becomes an urgent issue with VANET. The trust gained through neighboring nodes would enable them to acquire accurate and real-time information of accidents and any emergency situation on roads. It will also identify any denial of service (DoS) and intrusions on emergency activities that may have been encountered in the network. Moreover, safety applications such as SSVA, PCN, and CCA are connected with the RSU and are also deployed in VANET and connected at the traffic management office (TMO). However, the connection of these safety applications with the RSU requires improvement and efficient information delivery. The safety applications and the network devices can function appropriately and also ensure timely notifications about any accidents and road emergency situations. In addition, installation of VANET and appropriate deployment of the RSU with safety applications can help disseminate and process warning messages accurately in real-time without delay. Moreover, it is anticipated that warning messages can be conveyed in a timely manner through the VANET, through which the message can be relayed to other vehicles. The warning messages are usually generated at the TMO and may include notifications of DoS and intrusion attack activity in the network. Some of the DoS and intrusion attacks include congestion/collision (CC), link breakdown, and bad road conditions. CC of vehicles can occur in VANET at any time on the road due to the behavior of the disabled vehicle or accident which requires immediate attention and notification on a timely basis. In addition, some DoS intrusion attacks which this research investigates include smart and normal intrusions (SNI) attacks. DoS and SNI attacks may cause link breakdowns in the network. DoS and SNI attacks also overwhelm the network and block the entire V2V communication within VANET. DoS and SNI attacks encountered in VANET become road threats. When these occur, they prohibit VANET safety applications to function appropriately. In addition, they may lead to further attacks in VANET, including the bad road conditions and highway congestion encountering of many vehicles. This will also make it difficult for drivers to prevent road casualties in a timely manner. DoS, SNI, and DRA (DoS resilience attacker) all have the tendency to overwhelm the RSU. DoS and SNI can also exploit the RSU computational and communication resources and cause flooding with any requested information. However, the intent of RSUs and their deployed safety applications is to be able to collect and analyze the real-time information from vehicles. The information that is eventually received by V2V communication should be appropriately analyzed and evenly distributed to other neighboring vehicles, connected through VANET and safety applications on timely manner through the end-to-end (E2E) communication process. The E2E communications process in VANET is important; however, E2E communication may experience particular DoS and SNI attacks which can also overwhelm the RSU, which would then require urgent attention. The RSU may waste computational time, especially when it encounters false message or information. Therefore, the RSU requires an efficient and secure storage method to safeguard it from being compromised when delivering vehicle to roadside unit (V2RSU) and V2V messages in VANET [5].

In VANET, V2V and V2RSU communication storage solutions for propagating safety information to nearby vehicles in a timely manner have been investigated using vehicular cloud and fog computing (VCF) [6]. The VCF model has been developed to utilize VANET resources efficiently due to fog computing (FC) and cloud-based logical interaction. Based upon VCF, grouped vehicles cooperate and communicate with each other and dynamically share sensing, computation, and resources for decision-making on the road, as well as for improving traffic management and road safety. There are some examples of VCF applications that can be relied upon which include:

- Collecting local and highways traffic conditions from neighboring vehicles for planning routes.
- Processing the big data traffic information through local and highway traffic management authorities.
- Critical collaborative events including road congestion, accidents, and all forms of attacks (including DoS and SNI attacks) can be reconstructed.

Although these application scenarios have utilized FC and cloud-based applications for efficient storage and computations, this scheme has not been not appropriately secured. The authors claim that their proposed scheme has achieved their aim in investigating quality of service (QoS) parameters in VANET. Arguably, due to undetected DoS and SNI attacks, further investigation is needed. We believe fog computing (FC) integration and the hybrid deployment of optimization algorithms (OAs) including Cuckoo search algorithms (CSA), firefly algorithms (FA), firefly neural networks, and key distribution establishment (KDE)/authentication sharing mechanisms is a promising solution for investigating real-time data transmission and QoS parameters in VANET that answers to this question very well. Thus, we believe integration of the KDE/authentication mechanism investigation for the network level and the node level security can be achieved appropriately in order to ensure trustworthiness of nodes and trustworthiness for the entire VANET. In addition, since RSUs play a major role in distributing information in VANET, they can be secured appropriately to provide real-time end-to-end V2V and V2RSU communication. Therefore, it has become urgent to investigate QoS parameters such as delay/jitter and throughput in VANET. Moreover, due to the dynamic nature of VANET, it utilizes a vulnerable wireless link. Wireless link deployment and connection with vehicles and associates connect through multimedia safety applications should be secured when vehicles connect with the RSU [7]. Since multimedia safety applications are now a part of the VANET system, however, they are easily plagued by DoS and SNI attacks through the RSU. Multimedia safety framework demands high QoS support and evaluation. QoS provision, in general, is required to supports the Media Access Control (MAC) architectures [8]. MAC architectures for VANET rely on the VANET wireless medium which can be implemented on DSRC (dedicated short range communication) data link technology [9].

In the past, researchers/authors have conducted several investigations on VANET. The authors' investigation centered on multimedia safety application framework for determining QoS provision in VANET, which also utilized FC for achieving the network level security protection, using the DSRC data link technology. In addition, the authors have conducted separate investigations on OAs based upon FC while utilizing DSRC data link technology for data transmission. The authors' investigation involved CSA [10–12], FAs [13–15] and a firefly neural network [16]. The aim of the authors was to evaluate QoS parameters for delay/jitter and throughput in VANET. In addition, during the research investigation, a firefly neural network was used to train effective misbehavior of the path delayed in the VANET. Though the authors claimed to have succeeded investigating QoS performance in the network, the QoS evaluation was not complete due to the inability of the researchers/authors to consider the node level security evaluation in VANET. In addition, the authors did not investigate KDE sharing, including hybrid integration with OAs. Therefore, there was a limitation in the evaluation of trustworthiness in VANET, and both real-time information delivery and QoS provision within VANET remain a major concern. FC integration with OAs including KDE sharing can be useful for implementing VANET safety applications, since these schemes have the capability to ensure efficient storage, time sensitivity, trustworthiness, and intelligence in real-time information delivery agendas and QoS in Intelligent Transportation systems. To address these concerns, in this paper, we propose a "Secure Intelligent Vehicular Network using fog computing" (SIVNFC) scheme for FC integration and hybrid OAs deployment including CSA, FA, firefly neural networks, and KDE/authentication to detect the network level and node level security in VANET against DoS, SNI, and other forms of attacks.

The main contributions of this research are:

- Fog computing (FC) is integrated with hybrid OAs deployment including: CSA, FA, firefly neural networks, and KDE. FC is used to determine the rapidly stored vehicular information. In addition, the integration and deployment of FC with hybrid OAs and KDE provides intelligence which reduces the search space for real-time information. It also prevents increased communication times. Fog computing is an extension of cloud computing that provides computation, storage services, and network communication services between the end nodes. The determination of the rapidly stored vehicular information process relies on the communication behavior of vehicles in this paper [17].

- Secure the VANET at the node level and the network level for trustworthiness.
- Determine reduced jitter and improved throughput for the VANET for real-time data transmission.
- Use of regression model to confirm the accuracy of jitter/delay in the proposed SIVNFC scheme as a road safety application.

The organization of the rest of the paper is as follows. Section 2 presents related work. Section 3 discusses the DoS attacks, intrusions, and preventive mechanisms for the proposed SIVNFC model. Section 4 presents extensive simulation results and analysis of the results which includes: The feed forward-backward propagation neural network, regression model, and QoS provision for the VANET. Section 5 is the conclusion including the future work of this research.

2. Related Work

In this section, VANET-related work is divided into two subsections which include: Section 2.1: Securing VANETs-Centralized Architecture and Section 2.2: Securing VANETs-Fog Centric Distributed Architecture as follows.

2.1. Securing VANETs-Centralized Architecture

The architecture of VANETs and their operations are comprehensively analyzed in the literature [18]. The data sharing and key distribution mechanism during the data transfer were studied in [19]. Route discovery mechanisms were also developed and presented in the same scenario. We classify the security scenario at two levels: The security at the node level and the security at the network level. The node level security is applied when the selection of the node for the data transfer is involved, such as trusted node selection and the application of location-aware services [20].

In [21], the authors proposed location information verification cum security using a transferable belief model (TBM) for Geocast routing in VANET at the network level security. The proposed protocol included two level of location information verification. In the first level, tile-based techniques were used to verify location information correctness, whilst in level 2, collective information concerning the announced location information for each vehicle was obtained using TBM with the help of neighbor list information through all neighbor vehicles. The limitation of the proposed protocol is that it did not recommend any method for the network level security in order to evaluate trustworthiness in VANET. Rather, the proposed protocol only disputed traditional security methods and only proposed location information verification that was transferable in VANET. In addition, no appropriate storage solution was offered on a real-time data transmission scheme. The authors in [22] proposed a dynamic congestion control scheme (DCCS) for safety applications in vehicular ad hoc networks to determine only the network level security. The proposed scheme is a means whereby the reliable and timely delivery of data in safety applications can be ensured for road users and drivers. The proposed DCCS scheme objective also included the broadcasting of safety messages in order to ensure reliability and timely delivery of messages to all network neighbors. However, the disadvantage of the proposed scheme is that DCCS is without a fixed infrastructure. Moreover, there was no trustworthiness and efficient storage mechanism for the evaluation of real-time information in the network.

In [23], the authors proposed a location error resilient geographical routing (LER-GR) protocol for vehicular ad hoc networks to detect only the network level security. In the proposed LER-GR protocol, a Rayleigh distribution-based error calculation technique was utilized for evaluating error in location of neighbor vehicles. Based upon the LER-GR protocol, the least error location information was used for determining next forwarding vehicles. However, due to the dynamic mobility of VANET, the proposed protocol should have recommended an efficient storage solution and intelligence for data exchange in location information that would also ensure the reliability of data transmission. In addition, there was no trustworthiness evaluation to assess vulnerabilities in the network for secure transmission of location data. In [24], the authors proposed an algorithm that achieved secured time stable Geocast (S-TSG) for VANET in a vehicular traffic environment for only the network level security. The proposed protocol was intended to detect vulnerabilities including DoS attacks in VANET, due to a decentralized, open dynamic, as well as a limited bandwidth and control of overhead information. However, in the proposed protocol, there was no investigation conducted to evaluate either efficient storage or an intelligent and secure method solution in VANET for real-time data transmission. The protocol limitation also included an absence in optimize real-time vehicular traffic environment information processing. In [25], the authors proposed a geometry-based localization for GPS outage in a vehicular cyber physical system (VCPS) (GeoLV) for network level security protection only. The proposed localization technique was a GPS assisted localization which has the tendency to reduce location aware neighbor constraints in cooperative localization. In addition, the proposed GeoLV utilized mathematical geometry for estimating vehicle location and focused on vehicular dynamics and the trajectory of the road. Based upon the proposed scheme, static and dynamic relocations were performed to reduce the impact of a GPS outage on location-based services. However, the limitation of the proposed GeoLV technique was that it does not guarantee trustworthiness, and no FC method for efficient storage solution in VANET geometry-based localization for GPS outage in VCPS model was recommended or proposed in the scheme. It can be realized that the node level security detection was a major issue with the proposed schemes.

2.2. Securing VANETs-Fog Centric Distributed Architecture

Security at the network level is defined as when the data has to travel from the source to the destination. Secured routing, key distribution, and the encryption of data packets fall under the network security method. Fog computing is used to store the network data and to reutilize it to accelerate network performance. In [26], the authors introduced fog computing to extend cloud computing in the context of the middle fog layer among cloud and mobile devices and produce various benefits. The authors utilized a key sharing mechanism for secure transmissions. In [27], the authors further discussed the usage of fog computing by using an event-based data gathering scheme.

When a data transfer is called in the network, a node is summoned to perform some activity, and an event occurs. A route discovery process contains 'n' events, including attaching hops from the source to the destination. The addition of a hop also requires the identification of trustworthy nodes, which utilizes optimization algorithms (OAs) to perform a successful operation to help solve this type of issue in computer science [28].

This research paper specifically utilized a hybrid of optimized Cuckoo search algorithms (CSA) [10], firefly algorithms, [15] and firefly neural algorithms [16] to investigate DoS and SNI attacks. The investigation also detected the node level and network level security and mitigated the attacks for trustworthiness in VANET. In [12], the authors also conducted an investigation about the cognitive behavior of VANET for high-speed mobility of VANET. In the investigation, it was discovered that VANET also experienced frequent topology changes. In addition, it was discovered that VANET incurred memory storage challenges for allocating spectrum resources. Hence, in [12], the authors proposed the improved adaptive binary Cuckoo search algorithm to investigate DoS attacks in VANET. The researchers in [15] used the firefly algorithm to investigate vehicles that travelled along highways which encountered some form of VANET attacks. These vehicles that were deployed in the VANET were vulnerable due to DoS attacks which caused delays at the network level. Afterward, the authors utilized a clustering algorithm to facilitate good communication links. The authors' investigation centered on the real-time communication of the VANET to determine the efficiency of the messages for vehicles in order to receive traffic warnings in a timely manner. The authors' investigation conducted on the FA was also used to determine the reliability of the warning signals. The authors also conducted research in the FA and utilized the vehicles road-side infrastructure (RSU) regarding traffic safety warnings. In [16], the authors utilized the firefly neural algorithm, which is a combination of FA and a neural network, to investigate and train the VANET to determine the delay of the network. The parameters used for training the VANET was used to detect the network level DoS attacks, and the delay was evaluated in the network. The firefly neural algorithm utilized a machine learning process studied in VANET to determine the misbehavior of the vehicles/nodes for detecting DoS attacks. The model consisted of four main phases including data acquisition, data sharing, analysis, and decision making. Hybrid OAs deployment including CSA, FA, and the firefly neural network, can integrate with fog computing and KDE to determine the node level and network level security against DoS and SNI attacks.

Hybrid OAs deployments select the best solutions or minimize unnecessary solutions to retain the contrast of the objective function. OAs are either heuristic or metaheuristic in nature. The heuristic approach has problem-solving skills but is not suitable for each domain. NP-hard problems fall under heuristic optimization algorithms. Non-heuristic algorithms are adaptive in nature and may be applied for different sets of problems. More elaborately, the optimization can be further classified as Natural Computing, Swarm Intelligence, or Medical Computing. Both the CSA, FA and firefly neural network are classified as a Swarm Intelligence algorithms. There are various practices and architectures for the CSA, FA, and firefly neural swarm intelligence (SI) algorithms that relate to the CSA used in this research. In one practice or behavior, the Cuckoo bird lays its eggs in other birds' nest and leaves its eggs to be cared for by other bird species. In another behavior, a Cuckoo destroys all of its eggs, even if only one egg is damaged, due to it considering that the eggs are not suitable for further reproduction. In addition, this research paper has utilized the second behavior of the CSA in combination with Lagrange's method and the other swarm intelligence algorithms such as FA and firefly neural network to select trustworthy nodes and ensure that the entire VANET is secure. The description is given in the subsequent section. The network may suffer from different kinds of intrusions or attacks. One of the most common security threats is the Denial of Service (DoS) and the SNI. In [29], different structures of DoS attacks that also address the concern of SNI are discussed and presented. A detailed description of DoS and the SNI attacks are also given in Section 3 of this research paper.

3. DoS Attacks, Intrusions and Prevention Mechanism in VANET

3.1. DoS Attacks

VANET experience DoS attacks [30]. These attacks intercept the channel at the data link layer. DoS attacks are capable of bringing down the available network resources. Through DoS attacks, the VANET can be exploited through the RSU due to the following:

- Resources consumption: DoS attacks consume the available network bandwidth. They inject fake routing messages, resulting in congestion over the VANET. This degrades the end communicating entities performance and introduces jitter.
- Signal jamming: DoS attacks have a high tendency to jam the transmissions while using channel interference.
- Packet Drops: DoS attacks have a high tendency to drop all or any selected packets. This interrupts the routing process from the source to the destination communicating entities.
- The investigation of VANET security provisions, such as certificate-based identification and a authentication mechanism are beyond the scope of this research.

3.2. Attack Principles

Unlike wired architectures where the channel blockage or congestion is always due to the increased flow rates at links with bottlenecks, congestion in a VANET may occur due to the aggregation property of the vehicles. If the attacker densely aggregates his attacks near the victim, the attacker can occupy more communication channels [31]. The total transmission capacity of one node increases a linearly with the increase in the area. If the node count does not vary, then the hop capacity is O (k), where k is the node count of the network. The data transfer requires a route discovery, and the node count in a route may increase with the increase in the area. Each node has a probability of 1/k of interacting with the channel. There are m nodes that can act as attacking nodes such that the victim node has the likelihood of (1 − m/k) of interacting with the channel. Figure 1 illustrates the channel occupancy and interaction of the proposed model architecture [32].

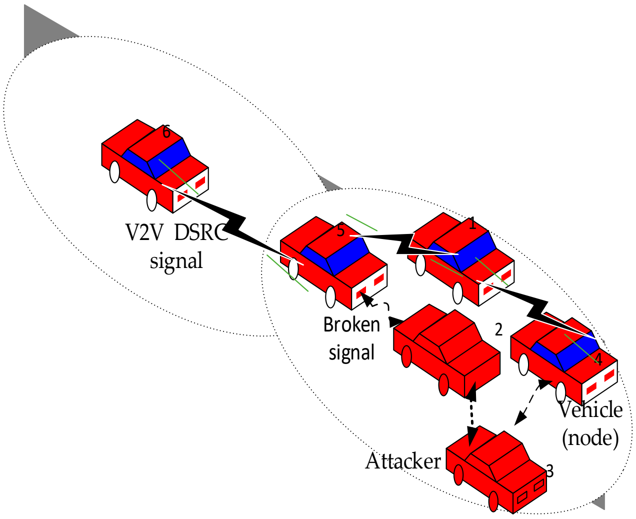


Figure 1. Channel occupancy.

VANET utilizes IEEE 802.11 as the most popular V2V DSRC (vehicle-to-vehicle dedicated short range communication) wireless system installed on almost every vehicle where the vehicle/channel congestion/collusion are inevitable due to influence of the attacker vehicle encounter in the network, which could occur at the time when vehicles (or V2V) are required to transmit packets to each other in VANET. CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance) is standard scheme that can be used to avoid such vehicle/channel packet transfer collision/congestion. However, CSMA/CA is only a simple mechanism that can be used to allocate radio resources. In this research, we investigated how vehicle/channel occupancy can cause delayed packet transmission due to misbehavior of attacker vehicle which leads to broken link exposure of the vehicles communication process as shown in Figure 1. Figure 1 illustrates channel occupancy scenario based upon the attacker mode of operation.

In Figure 1, there are two types of vehicles, namely attacker vehicle and normal vehicle. All attacker vehicles have broken signals connections with each other. When attacker vehicle forms a connection with normal vehicle, a delay can be experienced in the network due to channel occupancy as a result of broken signal connection because both normal vehicles and attacker vehicles are in each other's communication range and the vehicles are traveling on the highway. The first ellipse from left to right has a transmission range (250 m), whereas the next ellipse has an interference range (550 m). Attacker 2 transfers packets to vehicle node 3, and this processes are highlighted in a broken V2V DSRC communicating signal, in which the packet is not received by another normal corresponding vehicle. Now, vehicle nodes 5 and 4 are in the range of vehicle node 3, but since it is occupied by attacker 2, it will have to wait, and an unnecessary delay will occur in the network. The channel occupancy vehicular attacker scenario is also used to illustrate the misbehavior of compromised nodes in VANET due to DoS attacks [33].

3.3. DoS Attack Illustration

A DoS attack employs multiple vehicles to attain its goal. It locks the job queue of the corresponding vehicle so that it is unable to accept data packet requests from genuine vehicles. Since a DoS attack is distributed over several vehicles, distinguishing authentic users becomes complicated. There are several ways to mitigate the effects of this type of attack, including encryption and the use of classification techniques [30]. The use of authentication mechanisms can also be beneficial. Sanya Chaba et al. (2017) presented a VANET architectural design for authentication key delivery with less delay between vehicles and with more mobility by utilizing fog and cloud computing. The authors have also introduced fog computing to extend cloud computing to the context of the middle fog layer among cloud and mobile devices for the production of various benefits. In their work, Qi Jian et al. (2018) identified the security goals for VCC (vehicular cloud computing) interoperability.

The authors have provided the AKA (Authentication and Key Agreement) framework for VCC. Notably, the authors have proposed the problems with the challenges for designing a consistent AKA with extra strong security assurance for VCC. A hybrid AKA framework has been suggested that combines the 'single server 3-factor protocol' with the 'non-interactive identity-based key established protocol,' which computes the performance by a simulated platform. Fog computing is utilized quite often these days for deployment of VANET, but its implementation has not been deployed with any KDE or key sharing for preventing SNI attacks, also utilizing the RSU. Figure 2 illustrates an attack model scenario with the integration of the fog server with vehicles. In Figure 2 RSU stands for road side unit. The fog server keeps the information about the vehicles and distributes the required information to other vehicles if required. The intruder may also utilize the same server and may misuse the server's information to spread false information [33].

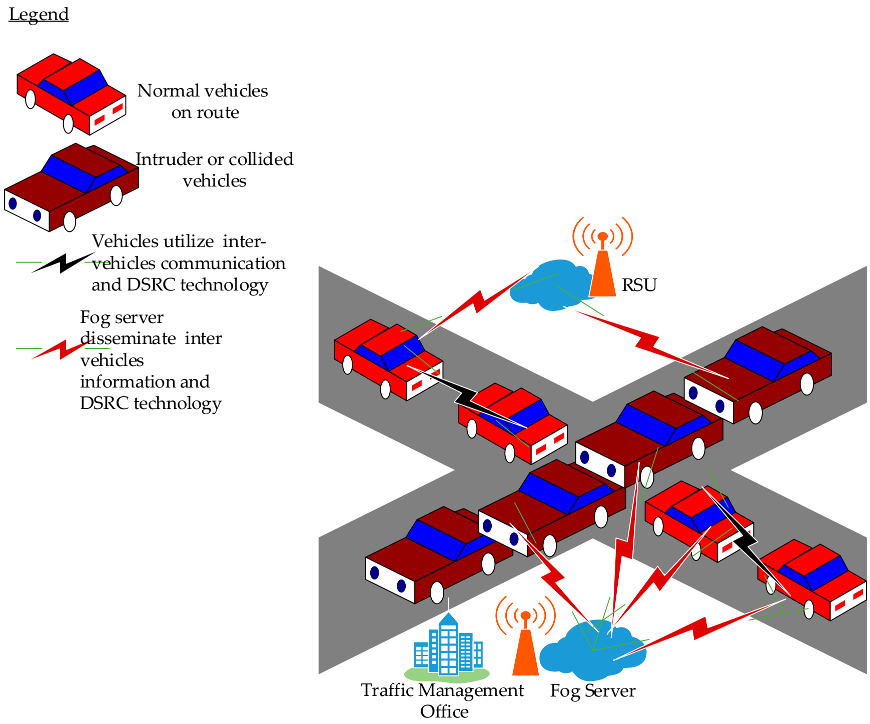

Legend
Normal vehicles on route
Intruder or collided vehicles
Vehicles utilize inter-vehicles communication and DSRC technology
Fog server disseminate inter vehicles information and DSRC technology
RSU
Traffic Management Office
Fog Server

Figure 2. Intrusion/attacker model.

3.4. *Intrusion /Attacks Model*

Figure 2 illustrate the intrusion/attacker model (IAM). The model detects and mitigates Dos and SNI attacks. The proposed IAM utilizes two types of vehicles, namely normal and intruder or collided vehicles. Normal vehicles are supposed to be on route. Normal vehicles denote all vehicles that have not experienced any form of attacks. Normal vehicles are the type of vehicles that are expected to arrive at their destination safely. The intruder or collided vehicles, on the other hand, are the type of vehicles that have encountered intrusion attacks. Normally they are not expected to arrive to their destination. Moreover, the intruder vehicles have the tendency to introduce delays in the network.

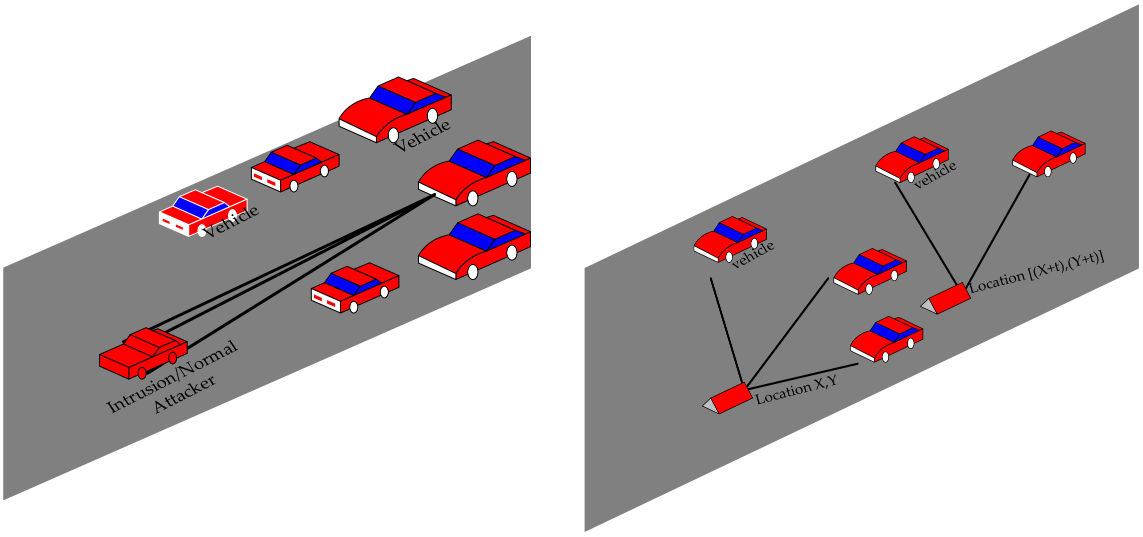
If intruder or collided/disabled vehicles are left unattended and continue to remain in the network, the network will suffer link breakdown and will not function as expected. This will lead to much delay encounter in the network. Delays of the network will lead to further road casualties since vehicles will not be appropriately informed. The proposed IAM initiates a remedy to prevent intruders/attackers in order to lessen road casualties. Therefore, in the proposed IAM, vehicles utilize intervehicle communication and DSRC technology. The vehicles communicate and share safety information with each other vehicle.

The information shared include condition of the vehicle and the road conditions. The information shared may also include congestion/collision and accidents that have already occurred. In addition, the fog server (FS) is deployed such that it addresses the location awareness concern in the cloud. The deployed FS disseminates emergency inter-vehicles information utilizing warning sign to alert other vehicles through the RSU information processing. The warning signal information can be obtained by each vehicle through the RSU and the FS which originates from the traffic management office (TMO). The TMO is the place where road safety applications (RSA) such including as SSVA, PCN, and CCA are deployed and connected with the RSU and the FS. Two inter-vehicle communications, including the FS, utilize DSRC technology. DSRC technology is data link technology which utilizes the IEEE 802.11 standard for transmitting information. Based upon this, real-time information, which convey warning and emergency information about any intruder activity in the network, can be received through the RSA.

The network is also identified with the other forms of intruder/attacker such as smart and normal intrusion (SNI). SNI may sometimes go unnoticed and requires sophisticated approach to detect. Smart intrusions make the network feel like there is no threat in the network. If the intrusion follows a set pattern of dumping the packets, then it becomes easy to identify. However, the smart intrusions do not follow a consistent pattern [34]. The SNI scenarios that occur in the VANET are depicted as in the figures below.

3.4.1. Smart and Normal Intrusion/Attacks Scenario

Figure 3a,b represent the normal and smart intrusions (SNI) attacker scenarios. The proposed IAM relies on the SNI intensity to evaluate the delay of the network. The intensity and location of the normal intrusion does not change with the change in the time frame, whereas the smart intrusion changes the location and intensity of the attacks with every instance. As shown in Figure 3b, the Intrusion is at location (x,y) at time t = 0, and it instantly changes its position at time t = 1 and goes to (x + t) and (y + t). The intrusion even changes the location and intensity of the attack at every instance [35]. The SIVNFC system architecture prevention mechanism (SAPM) is a sophisticated approach that can be utilized to determine and mitigate the SNI attacker in the VANET as demonstrated below.



(**a**) (**b**)

Figure 3. (**a**) Normal intrusion/attacker; (**b**) Smart intrusion/attacker.

3.5. Proposed SIVNFC System Architecture Prevention Model (SAPM)

Figure 4 depicts the proposed SAPM. In SAPM, vehicles utilize two DSRC technology instances for information transmission (the DSRC technology uses the IEEE 802.11 standard for transmitting information). In one instance of information transmission, vehicles communicate among themselves using intervehicle or V2V communication. In the other instance, the FS forms a connection with the RSU and, through this arrangement, disseminates inter-vehicle information to all vehicles in the network. The information conveyed usually include collusion/congestion, intruder activity of the network such as SNI of vehicles, or information of vehicles that have encountered attacks. The disseminated vehicle information may also include reporting the state of vehicles conditions and the road conditions that are threatening.

The proposed SAPM also employs further preventive measures to detect and mitigate all forms of attacks, including DoS attacks that may go unnoticed. Some of these attacks include but are not limited to packet drop, jamming of channels, and the RSU resources consumption overutilization. Two models are deployed in the SAPM, namely IAM and VANET structure with integrated for server (VSIF) models. The models utilize steps and scenarios for prevention and protection of the network against DoS and SNI attacks. In step 1, collided/disabled vehicles or intruder activity are detected and reported to the other vehicles in the network utilizing the IAM. The IAM detection of intruder/attacker has already been explained in detail above. In scenario 2, the VSIF model is deployed. The deployment of the VSIF model is also illustrated in Figure 5. The VSIF model relates and connect with the proposed SAPM as below.

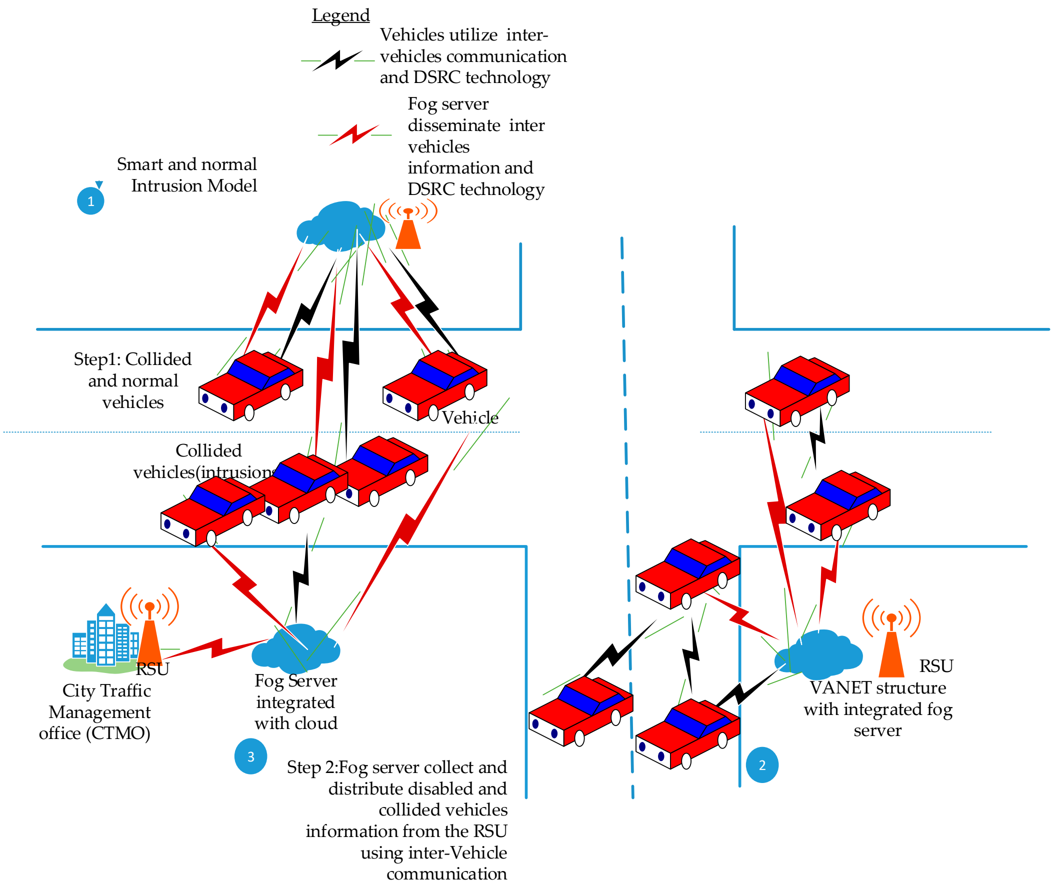

Legend
Vehicles utilize inter-vehicles communication and DSRC technology
Fog server disseminate inter vehicles information and DSRC technology
Smart and normal Intrusion Model
1
Step1: Collided and normal vehicles
Vehicle
Collided vehicles(intrusion
RSU
City Traffic Management office (CTMO)
Fog Server integrated with cloud
3
Step 2:Fog server collect and distribute disabled and collided vehicles information from the RSU using inter-Vehicle communication
RSU
VANET structure with integrated fog server
2

Figure 4. Proposed Secure Intelligent Vehicular Network using Fog Computing (SIVNFC) system architecture prevention model.

The VSIF model deployment in SAPM includes the RSU connection with the FS Step 3. Scenario 3 (Figure 4) illustrates the deployment of VSIF model, the FS, and the RSU connection. FS collects intruder or collided vehicles or any unusual network attack information. The FS also obtains information concerning all forms of DoS and SNI attacks that may be eminent in the network through the RSA which is installed at the TMO. The TMO is presumed connected with the RSU. The VSIF model utilizes inter-vehicle communications connections based upon the following deployment explanations.

RSU (Road side unit): RSUs are gateways. Gateways are also deployed in the proposed SAPM which establishes connections with the FS. The RSU is equipped with network devices. It utilizes DSRC inter-vehicle communication packet transfer based on IEEE 802.11.

RSU to FS: VANET utilizes V2V and V2RSU communication to propagate safety/non-safety information. RSUs communicates with each other as well. Thus, RSU behaves as the FS backbone. Wireless and wired connections are formed between RSU and FS (Figure 4). The RSU is aligned with FS.

Fog Server to Fog Server (FS to FS): FSs are identified at different locations. They interact with each other. Consequently, a pool of VANET resources that is localized can be managed through the TMO. This connection can be achieved via vehicular control center traffic management or TMO, as shown in Figure 4. Thus, direct wireless and wired communication between peer FS can be possible. In addition, collaborative services provision and the FS peer contents delivery can be initiated at the TMO, which improves the entire SAPM. In addition, the cloud is logically connected with the FS and has the tendency to aggregates information.

Fog Server to Cloud: In the proposed SAPM, FSs utilize fog computing to address location awareness concern of cloud computing. Thus, cloud computing represents a central portal of information which does not require location awareness for information processing. The cloud centrally controls the FS in various locations. A FS possesses the capability to aggregate the information that it has obtained from other FSs. The VSIF utilizes centralized computations whereby FS transmit intervehicle information that it has received from the cloud to the application users [36], utilizing the DSRC technology.

Due to open nature of the VANET deployment and associated vulnerabilities, RSU and the FS utilize an authentication/KDE preventive mechanism in the proposed SAPM for ensuring real-time packet delivery.

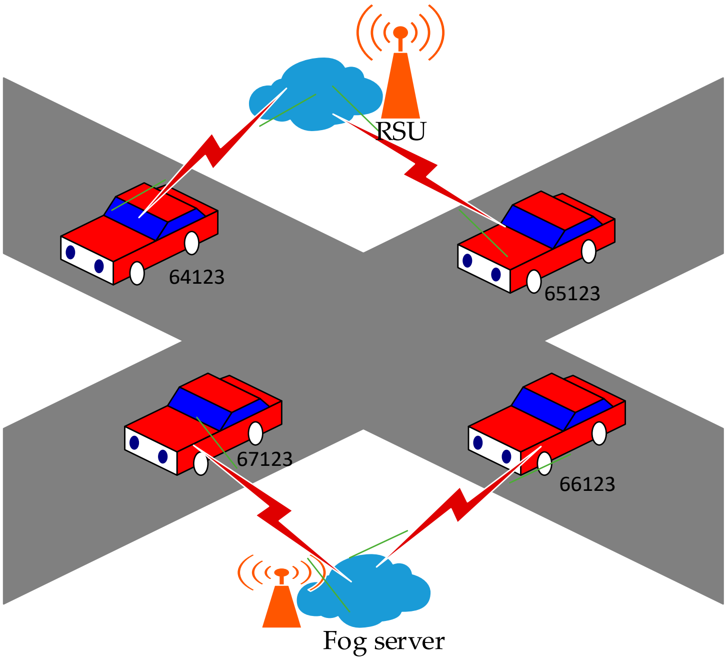


Figure 5. VANET (vehicular ad hoc network) structure with Integrated Fog Server Model.

The proposed SAPM utilizes two levels of authentication/KDE preventive mechanisms for the FS and the RSU aggregation of information, namely, the fog Level (FL) and the RSU Level (RSU-L).

The RSU-L considers the vehicle's displacement and jitter in the VANET, whereas the FL utilizes the Lagrange Polynomial for the identification of untrusted nodes as well [37].

3.5.1. FL Prevention Mechanism

The FL keeps one global key for the entire network; hence, each vehicle is identified by the global key itself. Distributing the global key in the vehicles is insecure; therefore, the vehicles follow a shared system. Each vehicle has its own shared value.

When a vehicle requests the information from a server either directly or through an RSU, the fog server will demand three shares from any vehicle in the network or will choose two of them randomly [38]. Three total shares will be considered, including the demanding vehicle. The fog server will utilize the Lagrange polynomial to calculate the following.

The Lagrange polynomial S(X) containing degree ≤ $(n-1)$ demands n vehicles with coordinates $\left(x_1, y_1 = f(x_1)\right), \left(x_2, y_2 = f(x_2)\right), \ldots\ldots\ \left(x_n, y_n = f(x_n)\right)$ is given by:

$$S(X) = \sum_{k=0}^{n} P_k(X) \tag{1}$$

where P_k is given by

$$P_k(X) = y_k \frac{x - x_l}{x_j - x_l} \text{ where } 1 \geq 1,\ l \leq n \text{ and } l\,! = k \tag{2}$$

If written explicitly for n = 3 vehicles,

$$S(X) = \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x1 - x_3)} y_1 + \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)} y_2 + \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)} y_3 \tag{3}$$

The separate polynomial can also be formulated as with Szeto (1975), which was later called Lagrange's fundamental interpolation.

$$S(X_1) = \frac{x_2 * x_3}{(x - x_2)(x - x_3)} y_1 \text{ for the first vehicle} \tag{4}$$

$$S(X_2) = \frac{x_1 * x_3}{(x - x_1)(x - x_3)} y_2 \text{ for the sec ond vehicle} \tag{5}$$

$$S(X_3) = \frac{x_1 * x_2}{(x - x_2)(x - x_3)} y_3 \text{ for the third vehicle} \tag{6}$$

The key that is generated by the integration of separate polynomials is represented as

$$G_k = \sum_{k=0}^{n} S(k) \tag{7}$$

If G_k matches the network key, only then does the vehicle pass any information from the fog server. Second, the RSU level security is also applied, which makes the network more secure. To understand the structure of this security, the pseudo code is also given as follows.

Pseudo Code Algorithm for Share Verification

```
Notations:
$SODFSV$: Shares Ordering Demanded by FS from Vehicles:
$I$ SVMy$_{\text{VALUE}}$[]: Initial Share for Vehicle Value being Empty
$SCV$: Share for Current Vehicles
$SKV$: Share Key Value
$SVCV$: Share Vehicle Current value
$V_i$: Individual $i^{th}$ Number of Share for Vehicle
$ICNSV$: Initial Counter Number for Share of Vehicle
$CSVBI$: Current Share for Vehicle/Node Begin Iteration
$SVNID$ : 1$^{st}$ Share Key Vehicle/Node Identification or Initial Reference
$SVNID_{Num}$: Share Numerator key for Vehicle/Node Identification in Network
$SVNID_{Deno}$: Share Denominator key for Vehicle/Node Identification in Network
$V_j$ : Individual $j^{th}$ Vehicle Chosen for Share in next Iteration
$SV_jC$: When first Vehicle Share is Chosen there will be 2 remaining Share for the Vehicle
$SVCNS$: Share for Vehicle Chosen Current not same as next Share Chosen
RSCV: Remaining Share Counter for Current Vehicle
Input. S(k), n, i, k, j,
Process
1. Initialization
$V_i = V_j$;
$ISVMy_{VALUE}[\,] = \varnothing$ ;
$SODFSV$=2 ;
$SVNID_{Deno} = \varnothing$;
2. If IS$VMy_{VALUE}! = \varnothing$;
3. for $V_i = 1 : 3$
While $ICNSV$ = =1; then
a. $CSVBI = SVNID;$
b. for $V_j == 1$;
c. CSVBI = $V_j$;
d. If $CSVBI$ ! = $V_j$.
e. $RSCV = SV_jC$ ;
4. $RSCV$= $RSCV$ +1;
5. End if
6. End for
8. $S\ VNID_{Deno} = V_j$-$\left(RSCV * V_j\right) - SV_jC$
9. $SVNID_{Num} = RSCV * SV_jC$
10. $ISVM$ My$_{\text{VALUE}}[i] = \frac{SVNID_{Deno}}{SVNID_{Num}}$
11. SKV= $SVCV$*$ISV$My$_{\text{VALUE}}[i]$
12. End for
Output : G$_k$, SCV
```

The pseudo code uses the interpolation order [39] of two and only three nodes for communication. Whether the nodes will be selected for the data communication or not depends upon the final key result, which is calculated using Lagrange's method. One key generation method requires a numerator and a denominator. The numerator is calculated using network IDs of the vehicles that remain for the iteration [40]. For example, we consider 45, 53, and 61 to be the nodes that are selected for the verification. Therefore, the numerator value (Num) for 45 is $53 * 61 = 3233$. The denominator (deno) is calculated by multiplying the difference of the network IDs of the remaining nodes. For 45, the deno value will be $(45 - 53) * (45 - 61) \rightarrow (-8) * (-16) \rightarrow 128$. The verification key would be the product of the Shared key of 45 to $\frac{\text{Num}}{\text{Deno}}$. Similarly, the Shared $_{\text{key}}$ for 53 and 61 will be calculated. The final verification key would be the sum of all the generated verification keys.

$$\mathrm{Final_{key}} = \sum_{k=0}^{i} \mathrm{My_{value}} \tag{8}$$

If the $\mathrm{Final_{key}}$ is equal to the network security key, then the nodes are selected for communication. Lagrange's theorem randomly selects the nodes for verification. Though the verification process of Lagrange is good enough, to make it more efficient, the CSA is applied to select the nodes for which the verification keys will be generated. The CSA uses the node distance and its feedback to judge whether it should be considered for key generation or not. Table 1 shows the specifications considered for the Cuckoo search algorithm (CSA).

$$\mathrm{LD} = \sqrt{\left((x_{nx1} - x_{nx2})^2 + \left(y_{ny1} - y_{ny2}\right)^2\right.} \tag{9}$$

Table 1. The specifications considered for the Cuckoo search algorithm (CSA).

CSA Population	Total Nodes in Coverage Region of Demanding Node
Fitness Parameters	Feedback, Location Difference (LD)

LD is the location difference between the demanding node and the communicating node. The CSA fetches the feedback values of nodes from the fog server, which also obtains intervehicle information through the RSU.

As shown in Figure 6, the main node computes the distance between the demanding node and the communicating node. It fetches the feedback from the fog server through the RSU and utilizes it for the fitness function.

$$\text{If Fitness}_{\text{function}} \rightarrow \quad \begin{array}{c} \text{Return 1 if} \frac{d}{f} \; \frac{\sum_{k=0}^{n} \frac{d_k}{f_k}}{n} \\ \text{Return 0 otherwise} \end{array} \tag{10}$$

where d is the distance between the fog and the user and f is the feedback of the fog server.

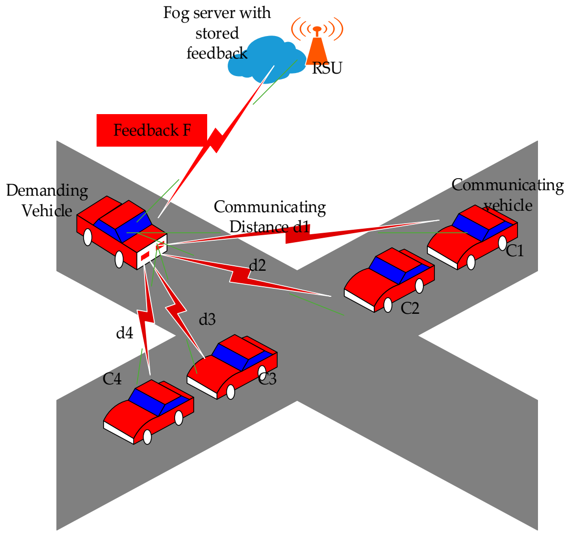


Figure 6. Node communication with a fog server.

The data transfer will take place once the route discovery process is complete. A network suffers from two kinds of security issues—namely, the node level and the data level. This paper further addresses the node level security [41].

3.6. Node level Security

VANET is a type of ad hoc network whose survival depends on vehicle/nodes cooperation and trust. Therefore, trust between vehicles requires enforcement. Trust models can be categorized into vehicle/node trust or data trust.

With node level trust security, vehicles/nodes evaluate trustworthiness between them, whereby each vehicle crosscheck their neighbors redundant sensing data with their results. Trust in vehicles can be calculated through a lightweight method and data which includes three parameters: Sensing a data consistency value (or throughput), VANET communication ability, and the Vehicle/nodes remaining lifetime. Trust assertion makes inconsistent data from DoS and SNI attacks to be detected [42].

The node level security is achieved by calculating the trust of neighboring nodes. The calculated trust values are stored in the fog server for further processing.

The mathematical equation for node level security in the VANET is calculated by determining the trust values of the node which is given as:

$$B = \mp \sum_{i=1}^{n} N_{xi}(Y) \tag{11}$$

The above equation shows that there is n number of trust factors. N(Y) indicates the trust value of the node of ith category. It is seen that if B is greater than or equal to N, the associated risk is less than threshold value and then node x will do work for Y. Node X keeps on checking to see any recommendations about Y node from neighboring nodes, and, if so, the trust value is calculated using the following equation.

$$C = \frac{\sum_{x=1}^{z} N_x(Y)}{z} \tag{12}$$

where z indicates number of neighboring nodes and $N_x(Y)$ indicates the trust value of node X on node Y.

The vehicles that have been identified as trusted nodes interact with the RSUs through the FS to obtain the data in the appropriate order [43]. The proposed SIVNFC scheme utilizes an RSU prevention mechanism whose model is as follows.

3.7. RSU-L Prevention Mechanism

The network deployment is based upon the specifications in Table 2.

Table 2. Network Specifications.

Total Number of Vehicles	**50–100**
Height of the Network	1000 m
Width of the Network	1000 m
Node Displacement	100–500 m/s
Simulation Iterations	1000
Simulation Tool	MATLAB

Pseudo Code for Vehicle Placement

```
    // To maintain the randomness in the network, the network is set in a random manner
For each n Nodes
Xloc(n)=1000*rand// Create a random x coordinate
Yloc(n)=1000*rand
lace(Xloc(n),Yloc(n))// Place the node in the network
End For
```

Vehicles have different sets of parameters. The functions are designed to initiate the network parameters. A real-time simulation may result in different structures. In addition, a network may not include any fixed structure; however, for the sake of any simulation, some parameters should be initialized.

Pseudo Code to Initialize Vehicle Features

```
For i=1:Nodes // Loop running for each node
Delay_n(i)=Random D; // Include a delay value if the node is acting normally
Delay_t(i)= Dealy_n$^2$; // For now, the expected reality is unpredictable; hence, just the random //architecture is it set to be the square of the normal delay
End for
```

As the delay is initialized in a similar fashion, the other network parameters such as the jitter and packet drop are also initialized. The battery consumption is not a problem in the case of a VANET since the battery continues charging as long as the vehicle is running [44].

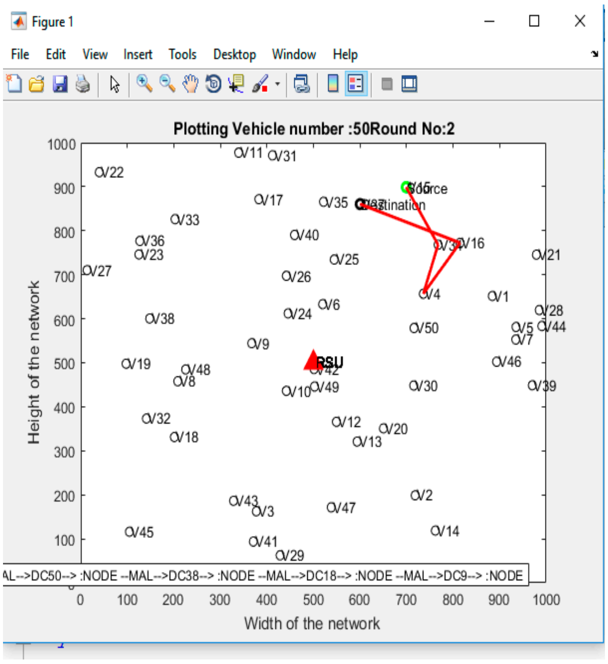
Figure 7a,b represents the path construction and attack mode of the attacker. Figure 7b shows that the intensity of the attacker varies at different times. If the intensity is high, the attacker is attempting to dump more packets. The above attacker scenario is demonstrated in the equations below.

$$\text{Tpd} = \text{Pdn} + \text{Pda} \tag{13}$$

where Tpd is the total packet drop, Pdn is the total number of dropped packets in the normal mode, and Pda is the dropped packets when the network is threatened.

$$\text{Pdr} = (\text{Tp} - \text{Tpd})/\text{Tp} \tag{14}$$

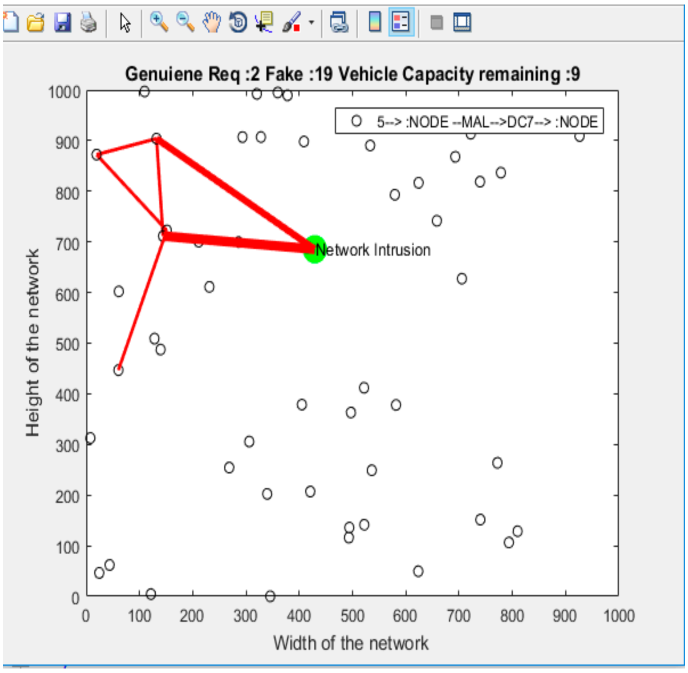
where Pdr is the packet delivery ratio, and Tp is the total number of packets. The random behavior of an attack makes the network architecture more sophisticated. Now, the challenge is to identify them. The proposed solution utilizes the feedforward back propagation neural network (FFBP-NN), and the general utilities of the FFBP-NN are given in Table 3.



AL-->DC50--> :NODE --MAL-->DC38--> :NODE --MAL-->DC18--> :NODE --MAL-->DC9--> :NODE

(**a**) Constructed path.

Figure 7. *Cont.*



(**b**). Attacker.

Figure 7. (**a**) Constructed path; (**b**) attacker.

Table 3. Utilized feedforward back propagation neural network (FFBP-NN) structure.

Total Hidden Layer	1
Neuron Count	30
Feeding Iteration	100
Reverse Iteration	40–60
Propagation Type	Linear
Algebraic Model	Levenberg

The Artificial Intelligence (AI) method is made up of two sections:

- Training and Classification

The classification section is used in the identification model. The training module utilizes the jitter as the training parameter. To train the neural network, the neural network toolbox in MATLAB is utilized. The training layer is provided with the target set as well. The target is the identification of the nodes. The training consists of two phases. First, training is performed for the identification of the path, and then the training is performed for the identification of the affected vehicle(s) in the route [45].

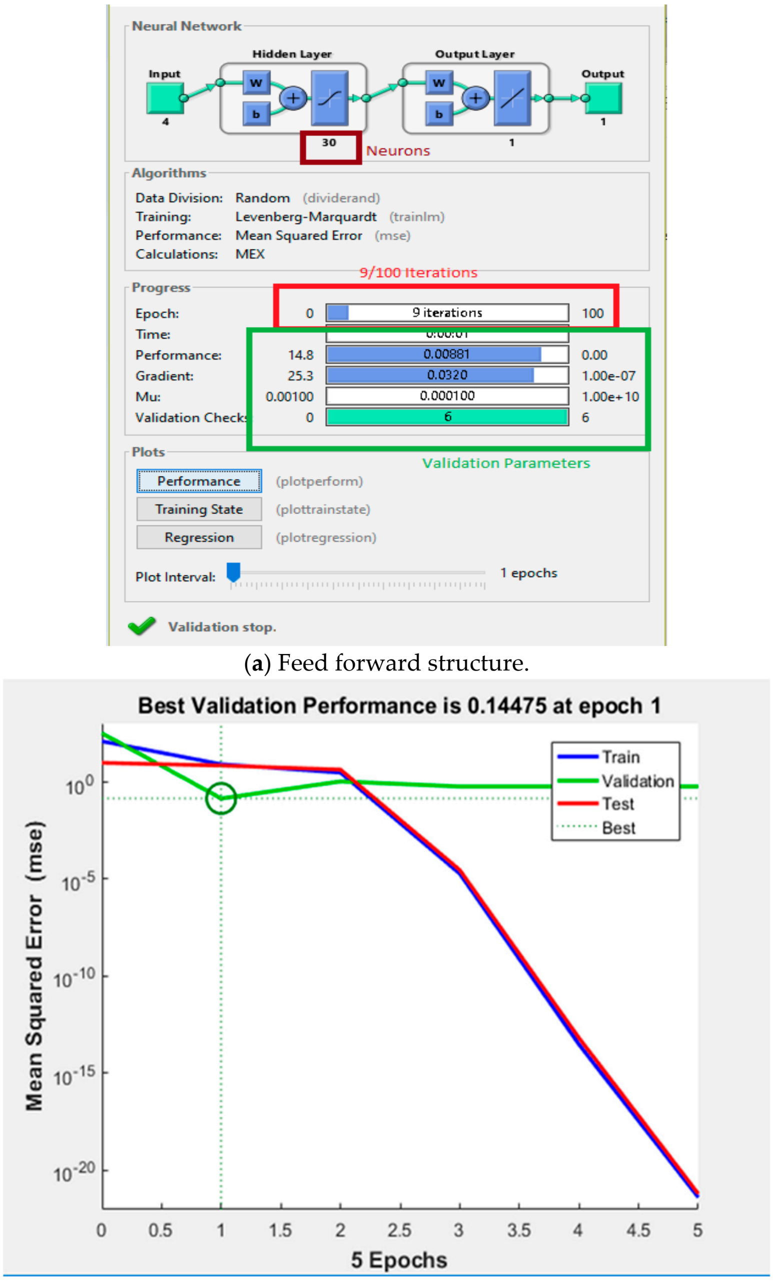
The following equation can be defined:

$$\text{Jtr} = \text{Dp}\ (\text{a}, \text{n}) + \text{Nd} \tag{15}$$

where Jtr is the jitter, Dp is the delay of the path, and ‘a’ and ‘n’ represent the advanced (under threat) and normal situations, respectively. Nd is the network delay. For each path in every iteration, there will be jitter. The proposed solution uses the first 400 iterations’ data for training and then uses the next 600 iterations’ data with the training structure for identification.

```
Algo Train_Neural (Iteration_Data,Total_Iterations)
   For i=1:Total_Iterations
 Training Data (i) =Iteration Data (i);
 Targetable (i) =Path ID;
 End For
 Neural=Initialize Neural (Training Data, Target Label, k); // k→ Total Neurons (30 in this case)
 NeuralI.TrainParam.Epochs=100; // Total training iterations
 Train (NeuralITraining_Data, Target_Label); // Training with Initialized Neural and Training data
 End Algorithm
```

The training section results in FFBP-NN structure given in Figure 8.



Algorithms
Data Division: Random (dividerand)
Training: Levenberg-Marquardt (trainlm)
Performance: Mean Squared Error (mse)
Calculations: MEX

9/100 Iterations

Progress
Epoch: 0 9 iterations 100
Time: 0:00:01
Performance: 14.8 0.00881 0.00
Gradient: 25.3 0.0320 1.00e-07
Mu: 0.00100 0.000100 1.00e+10
Validation Checks: 0 6 6

Validation Parameters

Plots
Performance (plotperform)
Training State (plottrainstate)
Regression (plotregression)
Plot Interval: 1 epochs

✓ Validation stop.

(**a**) Feed forward structure.

(**b**) Back propagation firefly.

Figure 8. (**a**) Feed Forward Structure; (**b**) Back Propagation Firefly.

3.8. *Identification of Affected Node(s) and Recovery*

The proposed research work also presents a regression model with backpropagation. Figure 9 represents the regression model and values.

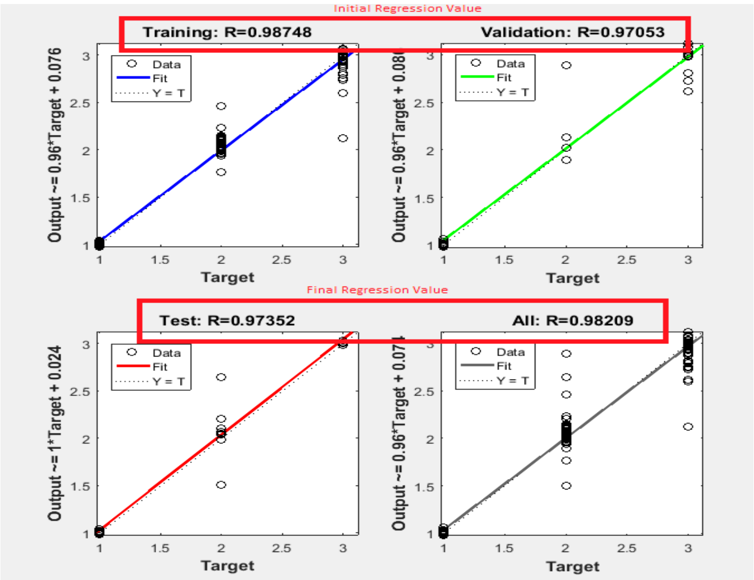


Figure 9. Regression model.

4. Results and Analysis

4.1. *Feed Forward–Backward Propagation and Regression Model Result and Analysis*

4.1.1. Feed Forward–Backward Propagation

From Figure 8, we can see that the proposed SIVNFC scheme calculates both the training data for latency (jitter) and validation of jitter that is the deviation between the predicted y and the actual y as a measure by the mean squared error (MSE). We can see that we have five Epochs for our model. This means that we are essentially training our model over five forwards and backwards. The five epoch is also the stopping iteration and the one epoch for back iteration. The expectation is that the proposed SIVNFC scheme will decrease with each epoch, which means that our model is predicting the value of y more accurately as we continue to train the model.

The predictions of the test data show how good the proposed SIVNFC scheme is. The test graph in Figure 8b, which indicates validation performance at epoch 1 of the model, indicates our model predictions is a good one.

From the graph in Figure 8b, we can see that both the training and the validation loss decreases in exponential fashion as the number of epochs is increased. This suggests that the model has gained high degree of accuracy as our epochs (i.e., the number of forward and backward passes) is increased.

4.1.2. Regression Model Result and Analysis

Figure 9 represents the close and high regression value of the proposed scheme. The result indicates that the proposed model close and high regression values are: Training is 0.98748, validation is 0.97053, test is 0.97357, and the value for all is 0.98209. All these regression values are close and high as well. Close and high regression values generally represent healthy training and classification structure. High regression value is the reason because of which the prevention parameters are high for the proposed model to prevent much jitter/delays in the SAPM architecture.

As discussed earlier, this section classifies the path value on the basis of the trained structure. The identified attacker nodes are always sent for recovery or maintenance.

4.2. QoS Provision Analysis in VANET

Development of VANET has recently received attention. Most of these attentions were based on the research effort conducted in the industry and in the field of academia [46]. VANET is classified as a key technology in intelligent transportation systems. VANET is envisaged as playing an important role in the futuristic smart cities. This important role in VANET improves road safety and also provide innovative services relating to traffic management and information achievement applications. Thus, it has become expedient for creating a wide range of services for future VANET deployment that ranges from safety/security and traffic management to commercial applications services [47]. Offering these services requires high QoS guarantees. Without QoS guarantees, these services would not be successfully achieved. Due to the highly dynamic nature of VANET, resources reservation for services are not applicable for providing a QoS guarantee.

In addition, two communicating vehicles that are moving would experience a degrading performance. This can be possible when the wireless links formed between them are vulnerable and the vehicles are disconnected due to DoS attacks. This can lead to unpredictable driver performance. QoS metrics such as throughput and jitter associated with the current routes established changes rapidly. The best selected routes computed by the RSU could easily become inefficient and lead to infeasible routes due to imminent links breakdown. Thus, utilizing a search for feasible route in multihop VANET is subject to multiple QoS constraints.

4.2.1. QoS Results and Analysis for the Proposed Model

The result and analysis of the proposed SIVNFC scheme is compared with the other contending models such as: CSA (Cuckoo), FA (firefly), and the firefly neural network. The analysis is based upon the QoS provision determination in VANET. The QoS analysis is based upon the simulation result and the mathematical analysis of the models in the SAPM. The QoS investigation is centered on throughput and jitter associated with the currents routes that has been established in the network as a result of rapid changes in the network due to the result of DoS and SNI in the VANET. We determine the QoS as follows:

- Throughput: It is the total number of delivered packets in the given time frame.

$$\text{Throughput} = \frac{\text{Total}_{\text{delivered}}}{\text{Time}_{\text{frame}}} \tag{16}$$

- Latency/jitter: It is the total delay that is produced when delivering data packets in the network.

The evaluation of the parameters is obtained in such a manner that the Packet Injection Rate (PIR) is on the x-axis and the QoS evaluation parameter is on the y-axis. The PIR is the ratio of the injection of the packets into the network.

Figure 10 demonstrates the results of the proposed SVINFC scheme, which is compared with all the other contending models. The proposed SINVNFC scheme considers the throughput with Cuckoo, firefly, and the firefly neural network. The range of PIR is from 0.001 to 0.02. With the increase in the PIR, the throughput increases, which is also demonstrated in Figure 10. The maximum throughput at PIR = 0.02 is 8100 for the proposed SIVNFC scheme and 7900 for the firefly–neural network odel. One hundred packets are injected per millisecond.

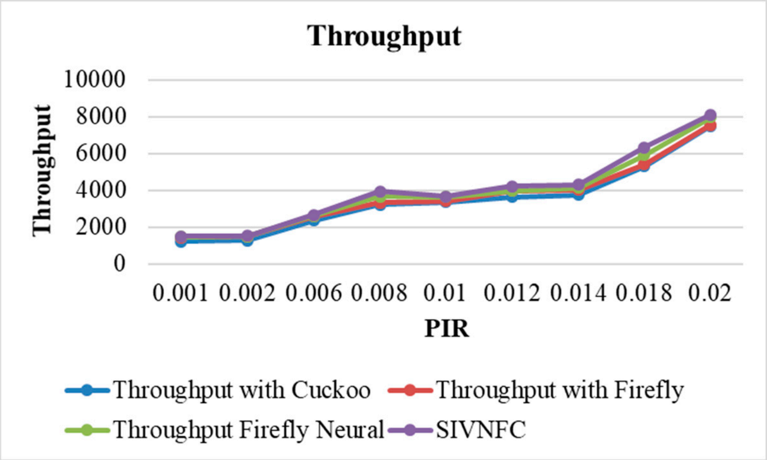


Figure 10. Throughput versus PIR.

The second evaluation parameter is the jitter. Jitter produces delays when the network experiences DoS and SNI. However, due to the fact that the proposed SVINFC scheme has introduced fog computing and that trust between the communicating neighboring nodes has been established, the entire network level security is increased. This has also led to decreased communication costs and time. The route that is discovered and assigned as trusted is stored on the fog server. Due to this, the need for broadcasting is reduced for route discovery and much time is saved. The evaluation of the jitter is done considering the same aspects as the throughput.

The jitter is not a consistent parameter in any network. Figure 11 shows that the jitter may be high or low for different PIR values. Throughout the PIR, the proposed SIVNFC scheme is noted to produce the least jitter when compared to other contending models scenarios. Though the fog computing server is applied to all the scenarios, the max jitter for the SIVNFC scheme is 96 ms, whereas the maximum jitter for Firefly Neural is 102 ms.

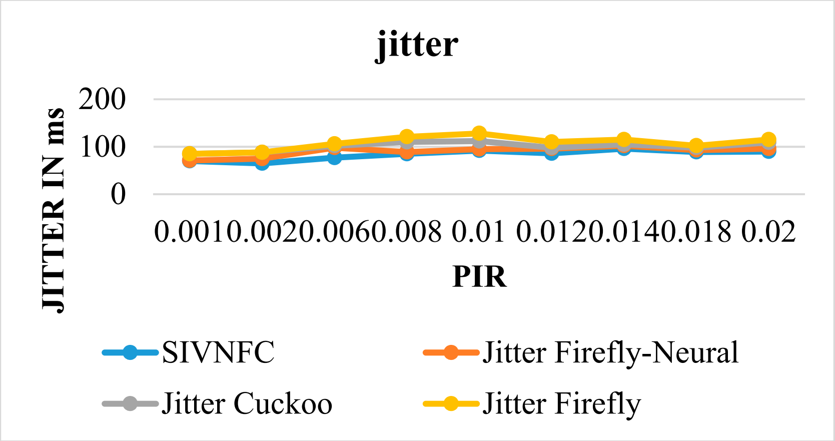


Figure 11. Jitter versus Packet Injection Rate (PIR).

Figure 12 represents close but least/high regression values of the proposed scheme. These results show detail regression model that was generated in the simulation before the final regression values were obtained. The result generated includes the following: The training result is 0.97847, the validation result is NaN (not a number), the test result is NaN, and the value for all result is 0.98727. Close and least/high regression values generally represent healthy training and classification structure as well, as indicated previously.

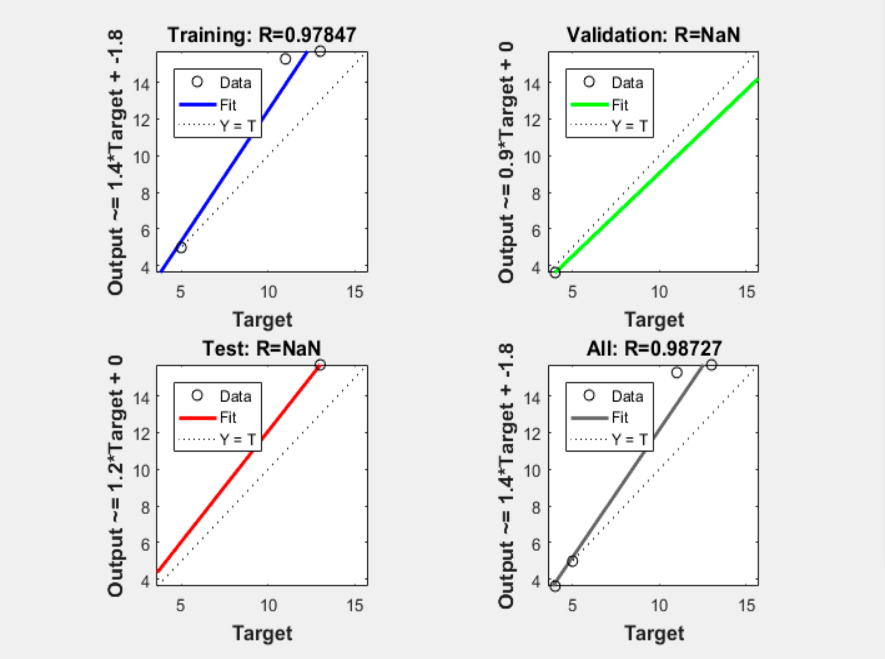


Figure 12. Detail regression model generated during simulations.

5. Conclusions

This paper proposed a fog-integrated VANET scheme termed SIVNFC. The proposed scheme simultaneously considers the node level and network level security. The node level security includes the fog computing merged with the VANET, a node level security mechanism, a new fitness function of the Cuckoo search, and a collaborated neural network structure. The node level security establishes trust collaboration with all the neighbors of the network. The node level trustworthiness ensures that the entire network rapidly delivers packets to the entire network system. The proposed SIVNFC scheme also prevent DoS attacks and SNI from attacking the entire network. The proposed SIVNFC scheme is an ad hoc network, and new vehicles, including DoS attacks, may easily enter into the network. To prevent the network from being accessed by foreigners (or outsiders) until they become part of the home network, the proposed SIVNFC scheme uses Lagrange's interpolation method through which node level and the network level security attacks such as DoS attacks entry is secured.

The proposed SIVNFC scheme also utilizes an integrated SAPM. The SAPM includes intrusion/attacker and VSIF models. Both models deployment in SAPM are utilized to mitigate all other forms of attacks and secure the network. The models are also deployed to provide real-time information in the network through safety application deployment of the RSU at the TMO, where information can be processed on timely to reduce delay and enhanced the throughput in the network. The evaluation of the proposed SIVNFC scheme is evaluated using QoS parameters—namely, the throughput and jitter. The proposed model is also compared with the firefly algorithm, a single neural network, a neural network combined with the firefly algorithm, and the Cuckoo Search algorithm.

The evaluation of the QoS parameters is done using the PIR as the basis of every simulation. The proposed SIVNFC scheme provided a total throughput of 8100 for the PIR value of 0.2. The maximum throughput of the network was also offered. For the same scenario, the second-best throughput was 7900 for the combination of Firefly and the neural network. The jitter is inconsistent throughout the simulations, and it varied based on the model architecture and algorithm. Even after nonlinear computations, the jitter for the SIVNFC scheme is a maximum of 96 ms, whereas it is 102 ms for the firefly neural network.

The proposed SIVNFC scheme also utilizes the regression model to indicate the reduced delay of the network. The current research work has potential for future research directions. The neural network structure can be varied to assess if there are any differences in the QoS parameters. A hybrid classifier can also be tested to see if it enhances the current proposed neural architecture. This paper utilized Lagrange's interpolation method, and it would be interesting to examine the performances of other interpolation methods such as Spline and the Polynomial fit. A combination of interpolation methods can also be considered.

Author Contributions: Conceptualization, S.K.E. and K.M.E.; Writing—original draft, S.K.E.; Writing—review editing, K.E.

Funding: This research has been funded by University of Bridgeport.

Acknowledgments: This research was developed and conducted by Samuel Kofi Erskine and Khaled M. Elleithy. The algorithms and formulas were solely developed by the authors. This research is sponsored by the University of Bridgeport.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Singh, D.; Ranvijay; Yadav, R.S. A state-of-art approach to misbehaviour detection and revocation in VANET: Survey. *Int. J. Ad Hoc Ubiquitous Comput.* **2018**, *28*, 77–93. [CrossRef]
2. Cooper, C.; Franklin, D.; Ros, M.; Safaei, F.; Abolhasan, M. A comparative survey of VANET clustering techniques. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 657–681. [CrossRef]
3. Hasrouny, H.; Samhat, A.E.; Bassil, C.; Laouiti, A. VANET security challenges and solutions: A survey. *Veh. Commun.* **2017**, *7*, 7–20. [CrossRef]
4. Sharma, S.; Kaul, A. A survey on Intrusion Detection Systems and Honeypot based proactive security mechanisms in VANETs and VANET Cloud. *Veh. Commun.* **2018**, *12*, 138–164. [CrossRef]
5. Panayappan, R.; Trivedi, J.M. *VANET-Based Approach for Parking Space Availability*; Carnegie Cylab Mellon University: Pittsburgh, PA, USA, 2007; pp. 1–4.
6. Grover, J.; Jain, A.; Singhal, S.; Yadav, A. Real-Time VANET Applications Using Fog Computing. In Proceedings of the First International Conference on Smart System, Innovations and Computing, Smart Innovation, Systems and Technologies, Jaipur, India, 15–16 April 2017; Springer Nature: Singapore, 2018; pp. 685–687.
7. Glass, S.; Mahgoub, I.; Rathod, M. Leveraging MANET-Based Cooperative Cache Discovery Techniques in VANETs: A Survey and Analysis. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 2640–2661. [CrossRef]
8. Chaba, S.; Kumar, R.; Pant, R.; Dave, M. Secure and efficient key delivery in VANET using cloud and fog computing. In Proceedings of the 2017 International Conference on Computer, Communications and Electronics (Comptelix), Jaipur, India, 1–2 July 2017; pp. 27–31.
9. Calandrelli, G.; Papadimitratos, P.; Hubaux, J.-P.; Lioy, A. *Efficient and Robust Pseudonymous Authentication in VANET*; Laboratory for Computer Communications and Applications, EPFL: Lausanne Switzerland, 2007; pp. 19–23.
10. Li, R.; Jin, L. Improved Cuckoo Algorithm for Spectrum Allocation in Cognitive Vehicular Network. In Proceedings of the 2018 5th International Conference on Systems and Informatics (ICSAI 2018), Nanjing, China, 10–12 November 2018; pp. 823–833.
11. Zhang, R.; Jiang, X.; Li, R. Decomposition based multiobjective spectrum allocation algorithm for cognitive vehicular networks. In Proceedings of the 2017 IEEE 17th International Conference on Communication Technology (ICCT), Chengdu, China, 27–30 October 2017; pp. 1–3.
12. Narawade, V.E.; Kolekar, U.D. EACSRO: Epsilon constraint-based Adaptive Cuckoo Search algorithm for Rate Optimized Congestion Avoidance and Control in Wireless Sensor Networks. In Proceedings of the 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 10–11 February 2017; pp. 715–720.
13. Azmat, F.; Chen, Y.; Stocks, N. Analysis of Spectrum Occupancy Using Machine Learning Algorithms. *IEEE Trans. Veh. Technol.* **2016**, *65*, 6853–6854. [CrossRef]
14. Sachdev, A.; Mehta, K.; Malik, L. Design of Protocol for cluster based routing in VANET using Fire Fly Algorithm. In Proceedings of the 2016 IEEE International Conference on Engineering and Technology (ICETECH), Coimbatore, India, 17–18 March 2016; pp. 1–3.

15. Kumar, R.; Chhabra, S. Efficient routing in Vehicular Ad-hoc Networks using Firefly optimization. In Proceedings of the 2016 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 26–27 August 2016; pp. 1–6.
16. Ghaleb, F.A.; Zainal, A.; Rassam, M.A.; Mohammed, F. An effective misbehavior detection model using artificial neural network for vehicular ad hoc network applications. In Proceedings of the2017 IEEE Conference on Application, Information and Network Security (AINS), Miri, Malaysia, 13–14 November 2017; pp. 1–5.
17. Agarwal, Y.; Jain, K.; Karabasoglu, O. Smart vehicle monitoring and assistance using cloud computing in Vehicular Ad Hoc networks. *Int. J. Transp. Sci. Technol.* **2018**, *7*, 60–73. [CrossRef]
18. Farhan, A.; Muhammad, K.; Asma, A.; Abir, A. Vehicular Cloud Networks Architecture, Applications and Security Issues. In Proceedings of the 2015 IEEE/ACM. International Conference on Utility and Cloud Computing (UCC), Limassol, Cyprus, 7–10 December 2015; pp. 571–576.
19. Vennila, R.; Duraisamy, V. Inter cluster communication and rekeying technique for multicast security in mobile ad hoc networks. *IET Inf. Secur.* **2014**, *8*, 234–239.
20. Kang, J.; Lin, D.; Jiang, W.; Bertino, E. Highly efficient randomized authentication in VANETs. *Pervasive Mob. Comput.* **2018**, *44*, 31–44. [CrossRef]
21. Dalya, K.; Omprakash, S.; Kaiwartya, O.; Abdullah, A.H.; Hassan, N. Location Information Verification cum Security using TBM in Geocast Routing. In Proceedings of the International Conference on Eco-Friendly Computing and Communication Systems, Haryana, India, 7–8 December 2015; pp. 219–221.
22. Qureshi, K.N.; Abdullah, A.H.; Kaiwartya, O.; Iqbal, S.; Butt, R.A.; Bashir, F. A Dynamic Congestion Control Scheme for safety applications in vehicular ad hoc networks. *Comput. Electr. Eng.* **2018**, *72*, 774–788. [CrossRef]
23. Kasana, R.; Kumar, S.; Kaiwartya, O.; Yan, W.; Cao, Y.; Abdullah, A.H. Location error resilient geographical routing for vehicular ad-hoc networks. *IET Intell. Transp. Syst.* **2017**, *11*, 450–452. [CrossRef]
24. Dora, D.P.; Kumar, S.; Kaiwartya, O.; Prakash, S. Secured Time Stable Geocast (S-TSG) routing for VANET. In Proceedings of the International Conference on Advanced Computing, Networking and Informatics, Smart Innovation, Systems and Technologies 2015, Bhubaneswar, India, 23–25 June 2015; pp. 161–162.
25. Kaiwartya, O.; Cao, Y.; Lloret, J.; Kumar, S.; Aslam, N.; Kharel, R.; Abdullah, A.H.; Shah, R.R. Geometry-based Localization for GPS Outage in Vehicular Cyber Physical Systems. *IEEE Trans. Veh. Technol.* **2018**, *67*, 3800–3801. [CrossRef]
26. Tangade, S.; Manvi, S.S.; Lorenz, P. Decentralized and Scalable Privacy-Preserving Authentication Scheme in VANETs. *IEEE Trans. Veh. Technol.* **2018**, *67*, 8647–8655. [CrossRef]
27. Lai, Y.; Zhang, L.; Wang, T.; Yang, F.; Xu, Y. Data Gathering Framework Based on Fog Computing Paradigm in VANETs. In Proceedings of the Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint Conference on Web and Big Data, Macau, China, 23–25 July 2017; Springer: Cham, Switzerland, 2017; pp. 227–236.
28. Shehab, M.; Khader, A.T.; Al-Betar, M.A. A survey on applications and variants of the cuckoo 797 search algorithm. *Appl. Soft Comput.* **2017**, *61*, 1041–1059. [CrossRef]
29. Liao, D.; Li, H.; Sun, G.; Zhang, M.; Chang, V. Location and trajectory privacy preservation in 5G-Enabled vehicle social network services. *J. Netw. Comput. Appl.* **2018**, *110*, 108–118. [CrossRef]
30. Singh, P.; Bart, W.N. Prevention of denial of service attack over vehicle ad hoc network using quick response table. In Proceedings of the 2017 4th International Conference on Signal Processing, Computing and Control (ISPCC), Solan, India, 21–23 September 2017; pp. 568–589.
31. Wang, M.; Liang, H.; Deng, R.; Zhang, R.; Shen, X.S. VANET based online charging strategy for electric vehicles. In Proceedings of the 2013 IEEE Global Communications Conference (GLOBECOM), Atlanta, GA, USA, 9–13 December 2013; pp. 4804–4809.
32. Giulio, M.; Lorenzo, V. On the performance of channel occupancy detectors for vehicular ad-hoc networks. In Proceedings of the 2013 IEEE 5th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Almaty, Kazakhstan, 10–13 September 2013; pp. 1–6.
33. Pathre, A.; Agrawal, C.; Jain, A. A novel defense scheme against DDOS attack in VANET. In Proceedings of the 2013 IEEE Tenth International Conference on Wireless and Optical Communications Networks (WOCN), Bhopal, India, 26–28 July 2013; pp. 1–5.

34. Bitam, S.; Mellouk, A. Bee life-based multi constraints multicast routing optimization for vehicular ad hoc networks. *J. Netw. Comput. Appl.* **2013**, *36*, 981–991. [CrossRef]
35. Pathre, A. Identification of malicious vehicle in vanet environment from DDOS attack. *J. Glob. Res. Comput. Sci.* **2013**, *4*, 30–34.
36. Whaiduzzaman, M.; Sookhak, M.; Gani, M. A survey on Vehicular cloud computing. *J. Netw. Comput. Appl.* **2014**, *40*, 325–344. [CrossRef]
37. Liu, J.; Li, J.; Zhang, L.; Dai, F.; Zhang, Y.; Meng, X.; Shen, J. Secure intelligent traffic light control using fog computing. *Future Gener. Comput. Syst.* **2018**, *78*, 817–824. [CrossRef]
38. Sookhak, M.; Yu, F.R.; Tang, H. Secure data sharing for vehicular ad-hoc networks using cloud computing. In Proceedings of the Ad Hoc Networks, Ottawa, ON, Canada, 28–30 June 2017; Springer: Cham, Switzerland, 2017; pp. 306–315.
39. Nobre, J.C.; de Souza, A.M.; Rosário, D.; Both, C.; Villas, L.A.; Cerqueira, E.; Gerla, M. Vehicular software-defined networking and fog computing: Integration and design principles. *Ad Hoc Netw.* **2019**, *82*, 172–181. [CrossRef]
40. Rauniyar, A.; Hagos, D.H.; Shrestha, M. A Crowd-Based Intelligence Approach for Measurable Security, Privacy, and Dependability. Internet of Automated Vehicles with Vehicular Fog. *Mob. Inf. Syst.* **2018**, *2018*, 828–829. [CrossRef]
41. He, Y.; Wei, Z.; Du, G.; Li, J.; Zhao, N.; Yin, H. Securing Cognitive Radio Vehicular Ad hoc Networks with Fog Computing. *Ad Hoc Sens. Wirel. Netw.* **2018**, 1–4.
42. Anbuchelian, S.; Lokesh, S.; Baskaran, M. Improving Security In Wireless Sensor Network Using TrustAnd Metaheuristic Algorithms. In Proceedings of the 2016 3rd International Conference on Computer and Information 859 Sciences (ICCOINS), Kuala Lumpur, Malaysia, 15–17 August 2016; pp. 233–238.
43. Wei, J.; Wang, X.; Li, N.; Yang, G.; Mu, Y. A Privacy-Preserving Fog Computing Framework for Vehicular Crowd sensing Networks. *IEEE Access* **2018**, *6*, 43776–43784. [CrossRef]
44. Islam, S.H.; Obaidat, M.S.; Vijayakumar, P.; Abdulhay, E.; Li, F.; Reddy, M.K.C. A robust and efficient password-based conditional privacy preserving authentication and group-key agreement protocol for VANETs. *Future Gener. Comput. Syst.* **2018**, *84*, 216–227. [CrossRef]
45. Sundarasekar, R.; Thanjaivadivel, M.; Manogaran, G.; Kumar, P.M.; Varatharajan, R.; Chilamkurti, N.; Hsu, C.H. Internet of Things with Maximal Overlap Discrete Wavelet Transform for Remote 839 Health Monitoring of Abnormal ECG Signals. *J. Med. Syst.* **2018**, *42*, 228. [CrossRef] [PubMed]
46. Yu, R.; Zhang, Y.; Gjessing, S.; Xia, W.; Yang, K. Toward cloud-based vehicular networks with efficient resource management. *IEEE Netw.* **2013**, *27*, 48–54. [CrossRef]
47. Eiza, M.H.; Owens, T.; Ni, Q.; Shi, Q. Situation-Aware QoS Routing Algorithm for Vehicular Ad Hoc Networks. *IEEE Trans. Veh. Technol.* **2015**, *64*, 5520–5522. [CrossRef]