Detecting RSU Misbehavior in Vehicular Edge Computing

Nalam Venkata Abhishek*, Teng Joon Lim*, Biplab Sikdar* and Ben Liang†
*Department of Electrical and Computer Engineering, National University of Singapore
†Department of Electrical and Computer Engineering, University of Toronto

***Abstract*—Mobile Edge Computing can be used to realize the low latency requirements of vehicular networks. However, by compromising the road side units (RSUs), an adversary can introduce an extra delay leading to various problems such as the wastage of edge computing resources and disruption of navigational and safety functions. The compromised RSU can for instance deliberately corrupt the PHY layer payload of the packets to be transmitted to the vehicles or drop the packets received from the vehicles. With such simple strategies, the adversary would increase latency and through that effect, create serious disruptions. Such attacks can affect many critical delay sensitive applications such as collision avoidance. To detect the presence of such an adversary, we propose a trust based detection system in this paper. Each vehicle transmits a feedback packet about every RSU it has interacted with to a central trusted server. Using the feedback obtained from multiple vehicles, at regular intervals, an aggregated trust value for each RSU in the network is obtained and is compared with a threshold to classify the RSU as authentic or malicious. We also present a mechanism to detect the presence of malicious vehicles reporting false feedback in the network. Simulation results demonstrate the effectiveness of the proposed detection mechanism and the impact of the choice of adversary parameters on the detection system.**

I. Introduction

Vehicular networks (VNETs) are mainly equipped with two types of communication; vehicle-to-vehicle communication (V2V) and vehicle-to-road side unit (or infrastructure) communication (V2I). It is necessary that communication systems operate with latency less than 50 ms to ensure high reliability [1]. With the aim of providing computing services in close proximity to the devices that need them, Mobile Edge Computing (MEC) is a potential solution to ensure reliability and low latency [2], [3]. MEC allows an autonomous vehicle to offload resource intensive operations and run applications on multiple platforms. The mobile edge hosts are generally installed on the road side units (RSUs) or can be located physically close to the RSU.

VNET, like any other network, are vulnerable to different types of attacks that compromise availability, confidentiality, authenticity, and integrity. Researchers in the past have proposed many methods to overcome such attacks [4], [5]. In all these cases, a compromised vehicle is the source of attack. However, RSUs are also vulnerable to cybersecurity attacks, or even physical damage due to their deployments along publicly accessible roads. Certain edge data centers (i.e. the RSUs in the case of VNETs) can be composed of microservers that lack the hardware protection mechanisms or include legacy edge devices with limited connectivity which restricts the authentication protocols that can be deployed [6], making it easier for an adversary to launch attacks. Possible attacks that can be launched by the adversaries include physical damage, service manipulation, privacy leakage, rogue data center, etc.

Few of these attacks are well investigated in the literature. A protocol for mutual authentication between the consumer and the provider (i.e. the mobile edge) to prevent the network from attacks like eavesdropping and man in the middle was presented in [7]. Rogue mobile edges can be detected using detection technique similar to the one presented in [8]. The detection scheme uses the round trip time between the user and the DNS server to detect rogue nodes. Jamming is another attack that could disrupt the wireless channel between the IoT devices and edge servers. Two algorithms, SAVE-S and SAVE-A, were presented in [9] to address stochastic jamming (i.e. where the IoT devices are attacked with a fixed probability) and adversarial jamming (i.e. where the attacker intelligently blocks certain edge servers). A reputation based trust management system to detect malicious cloudlets in LTE networks was presented in [10]. The system also prevents cloudlets from modifying the ratings from the mobile users and also limits the effect of dishonest ratings.

In this paper, we consider an adversary strategy where the adversary, after compromising the RSU, mimics a weak transmission link between the RSU and the vehicles. Such an attack is feasible once the attacker has obtained root access to the RSU. By doing so, the computation and communication latencies increase and create substantial additional delay in the network. This also leads to the wastage of edge computing resources. Hence there is need to design an Intrusion Detection System (IDS) for such attacks. A trust based IDS is presented in this paper to detect the presence of such an adversary. The key novelty behind the IDS proposed in this paper is that it is based on monitoring the downlink and uplink channels of the network. The IDS relies on the ability of the vehicles to measure the wireless channel quality, and the total number of packets received and transmitted. We also present a mechanism to detect the presence of any malicious vehicles reporting false feedback about authentic RSUs.

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its Corporate Laboratory@University Scheme, National University of Singapore, and Singapore Telecommunications Ltd. The research was partially supported by a Ministry of Education Tier 1 grant R-263-000-C92-114.

II. System Model

A vehicular network is considered where the vehicles are denoted using $V_j, j \in \{1, 2, \cdots\}$ and the road side units by $R_i, i \in \{1, 2, \cdots\}$. At a given instant, a vehicle V_j is connected to only one RSU say R_i. A Mobile Edge Host is installed on each RSU to enable MEC. All the RSUs are further connected to a central secured cloud server. For a given network in normal operation, due to various network imperfections, there will be a non-zero probability of decoding a packet in error. When a packet is not successfully received, a re-transmission request is made. The reception of a corrupted packet can be detected using the cyclic redundancy check (CRC). Unlike the case of static networks where the packet drop probability (PDP) can be assumed to be known, we need to obtain an estimate for the PDP of each packet separately in vehicular networks. We assume that a vehicle has access to a signal quality indicator to estimate the packet drop probability of every packet it receives. There are mainly three metrics that impact PDP: Signal to Noise Ratio (SNR), Link Quality Indicator (LQI) and Received Signal Strength Indicator (RSSI). In this paper, SNR will be assumed to be available and used to obtain the PDP.

We assume that the adversary has compromised RSUs and is disrupting the communication channel between the RSUs and the vehicles connected to them. A compromised RSU can be used to launch many attacks that target message security, quality of service etc. Attacks that focus on compromising message security by stealing credentials, eavesdropping, etc. can be handled using traditional algorithms based on cryptography. Attacks that focus on service intereption i.e. selective forwarding, black hole, jamming etc. have been thoroughly explored in the past. All these attacks can be dealt, with high reliability, using already proposed mechanisms in the literature [7]–[10] and therefore are not addressed in this paper. The adversary considered in this paper is implementing the Selective Modification Attack. The probability that a compromised RSU R_i selectively modifies a packet to be transmitted to the vehicle V_j is unknown and denoted by δ^d_{ij}. The adversary deliberately corrupts the PHY layer payload of a packet that needs to be forwarded to the vehicles. This can be achieved by corrupting the channel pilots that are used for channel estimation and/or flipping some of the bits of the physical layer payload. When such a packet is received by the vehicle, it will be dropped for failing error-control such as the CRC and a re-transmission will be attempted. Due to an increase in the number of redundant re-transmissions, the average delay in receiving a successful packet increases. Thus, the latency of the RSU-vehicle link is adversely affected and the safety of the network compromised. Such an attack is difficult to detect due to the dynamic nature of the wireless communication channel.

III. Intrusion Detection System – Downlink

In this section, we present the details of the trust based intrusion detection system. The IDS performs a binary hypothesis test on the RSUs independently:

- Hypothesis $H^d_{0,i}$: R_i is not malicious
- Hypothesis $H^d_{1,i}$: R_i is malicious

Under $H^d_{1,i}$, it appears that the wireless channel quality is poor although the SNR does not indicate such poor performance. Hence, the values of trust will be based on the SNR and the observed wireless channel quality.

A. Probability Distributions under both Hypothesis

For every packet $k, k \in \{1, 2, \cdots\}$, sent by RSU R_i and received by vehicle V_j, using the SNR of the packet, we estimate the probability $\alpha^d_{ij,k}$ that the packet could be in error. We first obtain the corresponding symbol error rate, denoted by $s^d_{ij,k}$, for the estimated SNR using standard models [11]. We then estimate the PDP as:

$$\alpha^d_{ij,k} = 1 - (1 - s^d_{ij,k})^m$$

where m is the number of symbols present in a packet. We introduce the variable $B^d_{ij,k}$, defined as follows. If the k^{th} packet is received successfully by V_j, then $B^d_{ij,k} = 0$, otherwise $B^d_{ij,k} = 1$. The probability distribution of $B^d_{ij,k}$ in the absence of attack (i.e. under $H^d_{0,i}$) is

$$P(B^d_{ij,k} = b | H^d_{0,i}) = (\alpha^d_{ij,k})^b (1 - \alpha^d_{ij,k})^{(1-b)}, b \in \{0, 1\}. \quad (1)$$

In the presence of an attack (i.e. under H_1), a packet is dropped either due to the RSU's misbehavior or due to poor channel conditions. Therefore the PDP of the k^{th} packet, in the presence of an attack, increases to

$$\beta^d_{ij,k} = \delta^d_{ij} + (1 - \delta^d_{ij})\alpha^d_{ij,k}. \quad (2)$$

The probability distribution of $B_{ij,k}$ in the presence of attack (i.e. under $H^d_{1,i}$) is therefore

$$P(B^d_{ij,k} = b | H^d_{1,i}) = (\beta^d_{ij,k})^b (1 - \beta^d_{ij,k})^{(1-b)}, b \in \{0, 1\}. \quad (3)$$

Since packet transmissions between R_i and V_j occur over a highly mobile channel, it can be assumed that the event of a packet being dropped is independent of another packet being dropped. Hence, the joint probability distribution of the variables $B_{ij,k}, k \in \{1, 2, \cdots, N_{ij}\}$, in the presence of attack, is given by the product of their individual probability distributions, as shown below.

$$P(B^d_{ij} = b^d_{ij} | H^d_{1,i}) = \prod_{k=1}^{N^d_{ij}} (\beta^d_{ij,k})^{b^d_{ij,k}} (1 - \beta^d_{ij,k})^{(1 - b^d_{ij,k})} \quad (4)$$

where $B^d_{ij} = \{B^d_{ij,1}, B^d_{ij,2}, \cdots, B^d_{ij,N^d_{ij}}\}$, $b^d_{ij} = \{b^d_{ij,1}, b^d_{ij,2}, \cdots, b^d_{ij,N^d_{ij}}\}$ and N^d_{ij} is the total number of packets received by the vehicle V_j from RSU R_i.

B. Detection Algorithm for RSU

Trust and reputation management has been proposed in recent years as an attractive method to deal with some security threats in highly distributed and dynamic scenarios, where there is no trusted central authority to directly monitor the network. We adopt this methodology in the VNET scenario for precisely that reason. Firstly, an individual vehicle's trust level towards the RSU would be calculated. This will be estimated at the vehicle's end. Secondly, at regular time intervals of duration T_1, the trust values obtained from all the vehicles about a given RSU would be combined (at the central server) to obtain an aggregated trust value for the RSU. Vehicle V_j's individual trust value for RSU R_i is denoted by Θ^d_{ij}. The aggregated trust value for RSU R_i is denoted by Θ^d_i. Once the aggregated trust is available, our hypothesis test decides in favour of $H^d_{1,i}$ if

$$\Theta^d_i < \Gamma^d_i \tag{5}$$

where Γ^d_i is a preset threshold. The proposed threshold setting method is discussed in the next few sections.

C. Individual Trust Evaluation

In this section, we compute Θ^d_{ij}, vehicle V_j's individual trust value for RSU R_i based only on downlink measurements. The trust value must dynamically reflect the behavior of the RSU. Trust, according to [12], is defined as the subjective probability by which an individual expects that another individual performs a given action on which its welfare depends. The key parameter which reflects the behavior of the RSU is the attack probability δ^d_{ij}. Its estimated value is denoted by $\hat{\delta}^d_{ij}$. This is obtained using maximum likelihood estimation (MLE) i.e. by maximizing the likelihood of $P(B^d_{ij} = b^d_{ij} | H^d_{1,i})$ over δ^d_{ij} between 0 and 1. It is difficult to obtain an analytical solution since the derivative of (4), when set to zero, results in the following equation:

$$\sum_{k=1}^{N^d_{ij}} \frac{b^d_{ij,k}}{\delta^d_{ij} + \alpha^d_{ij,k} - \delta^d_{ij}\alpha^d_{ij,k}} = \sum_{k=1}^{N^d_{ij}} \frac{1 - b^d_{ij,k}}{(1-\delta^d_{ij})(1-\alpha^d_{ij,k})}. \tag{6}$$

Therefore, we solve the problem numerically i.e. by exhaustive search between 0 and 1. Using this, we now define

$$\Theta^d_{ij} = 1 - \hat{\delta}^d_{ij}. \tag{7}$$

Since there is no analytical expressions for $\hat{\delta}^d_{ij}$, obtaining the distribution of $\hat{\delta}^d_{ij}$ is not possible. However, for sufficiently large values of N^d_{ij}, we can assume that the estimated attack probability i.e. $\hat{\delta}^d_{ij}$ can be approximated with its asymptotic probability distribution [13]. The asymptotic PDF of $\hat{\delta}^d_{ij}$ is:

$$\hat{\delta}^d_{ij} \sim \mathcal{N}(\delta^d_{ij}, {I^d_{ij}}^{-1}(\delta^d_{ij})) \tag{8}$$

where $I^d_{ij}(\delta^d_{ij})$ is the Fischer information [13] evaluated at δ^d_{ij}. In the absence of attack, the value of δ^d_{ij} is equal to zero. Using (8), the asymptotic PDF of Θ^d_{ij} under $H^d_{0,i}$ can be defined as:

$$\Theta^d_{ij} \sim \mathcal{N}(1, {I^d_{ij}}^{-1}(0)) \tag{9}$$

where $I^d_{ij}(0) = \sum_{k=1}^{N^d_{ij}} \frac{1-\alpha^d_{ij,k}}{\alpha^d_{ij,k}}$.

D. Aggregated Trust Value and Threshold Design

When there are more than one vehicle reporting trust values for an RSU, we require a way of combining the information. This process is referred to as aggregation. Trust aggregation helps in reducing the uncertainty of the detection algorithm. The widely used aggregation operators are minimum, maximum, weighted sum, and average [14]. It is quite possible that the trust values computed by some vehicles could be high even in the presence of an attack. This is mainly due to estimation errors in $\hat{\delta}^d_{ij}$. Therefore, when the maximum aggregation operator is used, there is a substantial probability that the system would classify the RSU as authentic. Similarly, it is possible that the erroneously obtained $\hat{\delta}^d_{ij}$ could be high in the absence of an attack implying a low trust value. In such cases, when the minimum aggregation operator is used, there is a substantial probability that the detection algorithm would classify the authentic RSU as malicious. Also, it is possible that different vehicles receive different numbers of packets from the RSU to which they are connected. Therefore, using a weighted mean with different weights for different vehicles would be more suitable to obtain the aggregated trust. Let $\mathcal{V}_i$, with $|\mathcal{V}_i| = M$, be the set of vehicles whose trust values are used for computing Θ^d_i, the aggregate trust value for R_i. Using the downlink individual trust values reported by the vehicles $V_j, j \in \mathcal{V}_i$, the aggregated trust value Θ^d_i is defined as

$$\Theta^d_i = \sum_{j \in \mathcal{V}_i} \omega^d_{ij} \Theta^d_{ij} \tag{10}$$

where ω^d_{ij} is the weight of the vehicle $V_j, j \in \mathcal{V}_i$. A higher number of packets N^d_{ij} implies the attack probability can be estimated with better accuracy. Therefore, we assign the weights in proportion to the number of packets transmitted from R_i to V_j.

$$\omega^d_{ij} = \frac{N^d_{ij}}{\sum_{j=1}^{M} N^d_{ij}}. \tag{11}$$

Since the system is time varying, we propose to design the threshold adaptively, i.e. obtain a different threshold for different sets of trust values reported. To obtain the threshold, we first obtain the values $\Gamma^d_{ij}, j \in \mathcal{V}_i$ such that the false alarm probability $P(\Theta^d_{ij} < \Gamma^d_{ij} | H^d_{0,i}) = \mu_d$ for $j \in \mathcal{V}_i$ where μ is a design parameter. The expression for Γ^d_{ij}, using (9), is given by

$$\Gamma^d_{ij} = \max(\min({\Gamma'}^d_{ij}, 1), 0) \tag{12}$$

where ${\Gamma'}^d_{ij} = 1 + \frac{Q^{-1}(1-\mu_d)}{\sqrt{I^d_{ij}(0)}}$. $Q^{-1}(x)$ is the inverse of the tail distribution function of the standard normal distribution. Using the obtained values of $\Gamma^d_{ij}, j \in \{1, 2, \cdots, M\}$, we set the threshold as:

$$\Gamma^d_i = \sum_{j \in \mathcal{V}_i} \omega^d_{ij} \Gamma^d_{ij}. \tag{13}$$

IV. Intrusion Detection System – Uplink

In this section, we consider an attacker who is implementing the selective dropping attack on uplink traffic. The compromised R_i drops an uplink packet from vehicle V_j with probability δ^u_{ij}. The effect of implementing such an attack is similar to the selective modification attack i.e. the latency is increased and safety of the network is compromised. Similar to the selective modification attack, the attacker startegy presented in this section is difficult to detect due to the nature of the wireless communication channel. Therefore to detect such an attack, we perform a binary hypotheses test with the following hypotheses:

- Hypothesis $H^u_{0,i}$: R_i is not malicious
- Hypothesis $H^u_{1,i}$: R_i is malicious

A. Packet Drop Probability

It was possible to estimate the probability of dropping downlink packets, at the vehicles, since they are received by them. In the case of the attacker strategy presented in this section, the uplink packets are affected. These packets are received by the RSU and monitoring of uplink packets cannot be done in the same manner as in the downlink. Assume that each vehicle has knowledge of its own location and speed, and the RSU's location. Using these three parameters we can estimate the quality (in terms of symbol error probability) of the wireless link between the vehicle and the RSU using standard channel models. We can then calculate the PDP of the k^{th} packet, transmitted by V_j and received by R_i, as follows:

$$\alpha^u_{ij,k} = 1 - (1 - s^u_{ij,k})^m \tag{14}$$

where $s^u_{k,ij}$ is the symbol error probability of the k^{th} packet.

B. Individual Trust Values

Similar to our previous work in [15], we use the packet re-transmission rate to obtain the trust values. We introduce the variable $B^u_{ij,k}$ for the k^{th} packet, transmitted by V_j, which is defined as follows. If a request for re-transmission is made (explicitly or implicitly) for the k^{th} packet by R_i, then $B^u_{ij,k} = 0$, otherwise $B^u_{ij,k} = 1$. The probability distribution of $B^u_{ij,k}$ in the absence of attack is

$$P(B^u_{ij,k} = b|H^u_{0,i}) = (\alpha^u_{ij,k})^b(1 - \alpha^u_{ij,k})^{(1-b)}, b \in \{0, 1\}. \tag{15}$$

In the presence of an attack (i.e. under H_1), a packet is dropped either due to the RSU's misbehavior or due to poor channel conditions. Therefore the uplink PDP of the k^{th} packet, in the presence of an attack, increases to

$$\beta^u_{ij,k} = \delta^u_{ij} + (1 - \delta^u_{ij})\alpha^u_{ij,k}. \tag{16}$$

The probability distribution of $B^u_{ij,k}$ in the presence of attack (i.e. under H_1) is therefore

$$P(B^u_{ij,k} = b|H^u_{1,i}) = (\beta^u_{ij,k})^b(1 - \beta^u_{ij,k})^{(1-b)}, b \in \{0, 1\}. \tag{17}$$

Similar to the downlink case, the packet transmissions between R_i and V_j occur over a highly mobile channel and can be assumed to be independent. Hence, the joint probability distribution of the variables $B^u_{ij,k}, k \in \{1, 2, \cdots, N^u_{ij}\}$, in the presence of attack, is given by the product of their individual probability distributions, as shown below.

$$P(B^u_{ij} = b^u_{ij}|H^u_{1,i}) = \prod_{k=1}^{N^u_{ij}} (\beta^u_{ij,k})^{b^u_{ij,k}}(1 - \beta^u_{ij,k})^{(1-b^u_{ij,k})} \tag{18}$$

where $B^u_{ij} = \{B^u_{ij,1}, B^u_{ij,2}, \cdots, B^u_{ij,N^u_{ij}}\}$, $b^u_{ij} = \{b^u_{ij,1}, b^u_{ij,2}, \cdots, b^u_{ij,N^u_{ij}}\}$ and N^u_{ij} is the total number of packets transmitted by the vehicle V_j to RSU R_i. The vehicle V_j's uplink individual trust value, Θ^u_{ij}, can now be calculated similar to Θ^d_{ij} since the probability distributions of B^u_{ij} under both the hypotheses are similar to B^d_{ij}. Therefore, the trust value is as follows

$$\Theta^u_{ij} = \delta^u_{ij} \tag{19}$$

where δ^u_{ij} is the MLE obtained by maximizing the likelihood of $P(B^u_{ij} = b^u_{ij}|H^u_{1,i})$ over δ^u_{ij} between 0 and 1. Also, the PDF of Θ^u_{ij} under $H_{0,i}$ can be defined as:

$$\Theta^u_{ij} \sim \mathcal{N}(1, {I^u_{ij}}^{-1}(0)) \tag{20}$$

where $I^u_{ij}(0) = \sum_{k=1}^{N^u_{ij}} \frac{1-\alpha^u_{ij,k}}{\alpha^u_{ij,k}}$.

C. Aggregated Trust Value and Detection Algorithm

Using the individual trust values reported by the vehicles $V_j, j \in \mathcal{V}_i$, the aggregated trust value Θ^u_i is defined as

$$\Theta^u_i = \sum_{j \in \mathcal{V}_i} \omega^u_{ij}\Theta^u_{ij} \tag{21}$$

where ω^u_{ij} is the weight of the vehicle $V_j, j \in \mathcal{V}_i$. A higher number of packets N^u_{ij} implies the attack probability can be estimated with better accuracy. Therefore, we assign the weights in proportion to the number of packets transmitted from V_j to R_i.

$$\omega^u_{ij} = \frac{N^u_{ij}}{\sum_{j=1}^{M} N^u_{ij}} \tag{22}$$

Once the aggregated trust is available, our hypothesis test decides in favor of H_1 if

$$\Theta^u_i < \Gamma^u_i \tag{23}$$

where Γ^u_i is a preset threshold. To obtain the threshold, we first obtain the values $\Gamma^u_{ij}, j \in \mathcal{V}_i$ such that the false alarm probability $P(\Theta^u_{ij} < \Gamma^u_{ij}|H_{0,i}) = \mu_u$ for $j \in \mathcal{V}_i$ where μ is a design parameter. The expression for Γ^u_{ij}, using (9), is given by

$$\Gamma^u_{ij} = \max(\min({\Gamma'}^u_{ij}, 1), 0) \tag{24}$$

where ${\Gamma'}^u_{ij} = 1 + \frac{Q^{-1}(1-\mu_u)}{\sqrt{I^u_{ij}(0)}}$. $Q^{-1}(x)$ is the inverse of the tail distribution function of the standard normal distribution. Using the obtained values of $\Gamma^u_{ij}, j \in \{1, 2, \cdots, M\}$, we set the threshold as:

$$\Gamma^u_i = \sum_{j \in \mathcal{V}_i} \omega^u_{ij}\Gamma^u_{ij}. \tag{25}$$

V. Detecting Malicious Vehicles and Feedback Packet Design

In this section we present a mechanism to detect the presence of malicious vehicles reporting false (downlink) feedback. The main objective of these vehicles is to influence the IDS into classifying an authentic RSU as malicious. Say a malicious vehicle V_j is trying to make it appear that the attack probability of the RSU R_i (which is authentic), i.e. δ_{ij}, is equal to δ_v. The feedback is generated as follows:

1) When the k^{th} packet is received, irrespective of whether it is in error or not, the packet drop probability is calculated (say $\alpha^d_{ij,k}$). To generate the false feedback for this packet, the PDP is reported as $\beta^d_{ij,k} \triangleq \alpha^d_{ij,k} + \delta_v - \alpha^d_{ij,k}\delta_v$.
2) Then, using a Bernoulli distribution with probability $\beta^d_{ij,k}$, the vehicle decides the value of $B^d_{ij,k}$. This is required to compute the Fischer information.

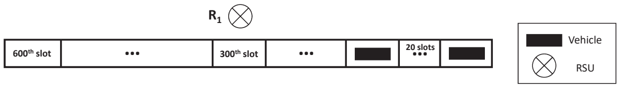

Fig. 1: Simulation model

There is a need to identify such malicious vehicles since they can influence the IDS. The key observation used is that the trust values of the RSUs in the absence of attack will to be greater than the individual trust values reported by the malicious vehicles, which implies that the distance between $L^d = \{\Theta_1^d, \cdots, \Theta_K^d\}$ and $L_j^d = \{\Theta_{1j}^d, \cdots, \Theta_{Kj}^d\}$ will be large if V_j is giving false feedback. Therefore, to detect such vehicles, we need a metric which can calculate the similarity between the L^d and L_j^d. One possible metric is the Gaussian kernel similarity measure [16]. For V_j, it is calculated as:

$$\rho_j^d = \exp(-\left\|L^d - L_j^d\right\|^2). \tag{26}$$

We now use this value of ρ_j^d and compare it with a preset threshold γ_j^d to detect whether vehicle V_j is malicious or not, i.e. we decide that the (downlink) feedback is false if and only if

$$\rho_j^d < \gamma_j^d. \tag{27}$$

This will be evaluated at regular time intervals, denoted by T_2. Similarly, malicious vehicles reporting false (uplink) feedback can be detected i.e. we decide the (uplink) feedback is false if and only if

$$\rho_j^u < \gamma_j^u \tag{28}$$

where $\rho_j^u = \exp(-\left\|L^u - L_j^u\right\|^2)$, $L^u = \{\Theta_1^u, \cdots, \Theta_K^u\}$ and $L_j^u = \{\Theta_{1j}^u, \cdots, \Theta_{Kj}^u\}$.

The feedback packet that needs to be reported by V_j for RSU R_i needs to contain the following information. The individual trust values, the total number of packets received and transmitted are required for calculating the aggregated trust values. The Fischer information $I_{ij}^d(0)$ and $I_{ij}^u(0)$ also need to be estimated and reported for calculating the threshold value.

VI. Results

It can be observed from (5) and (23) that the detection algorithms for unicast uplink and downlink packets are similar. Hence, in this case, we present the results for the unicast case only. We used MATLAB to generate the results. The model shown in Figure 1 is considered. The length of the road considered is 300 meters and the RSU, denoted by R_1, is placed at its mid point. The road is divided into 600 slots. If a vehicle is present in the U^{th} slot, the distance between the RSU and the vehicle is given by $|U - 300|/2$. In every time slot, the vehicle moves one slot. A new vehicle arrives in every 30 time slots. Hence, at any time there are 20 vehicles on the road. A new packet is transmitted per time slot. Using a uniform distribution, in MATLAB, we decide which vehicle is transmitting. The path loss model in [17] is used where $L_0 = 47dB$ and $x = 3$. Additive noise variance is $-100dBm$. The transmit power is equal to $20dBm$. Using the symbol error rate expressions in [11] for 16 QAM transmission in Rayleigh fading channels, the PDP is calculated.

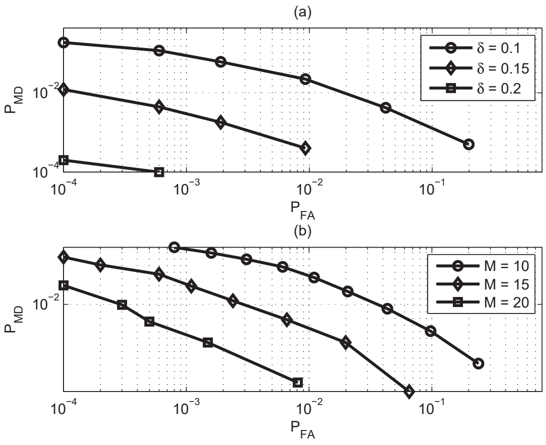

Fig. 2: Performance of the IDS when (a) M = 20 and the value of δ is varied (b) $\delta = 0.15$ and the value of M is varied.

The performance of our detection system is characterized using false alarm and missed detection probabilities, denoted by P_{FA} and P_{MD} respectively. The probability that the detection system decides on H_1 in the absence of an attack is defined as the false alarm probability. The probability that the detection system decides on H_0 in the presence of an attack is defined as the missed detection probability. To obtain P_{FA}, we setup the network using Hypothesis H_0 i.e. $\delta_{1j}^d = 0\ \forall j$. In every iteration, using simulation, we obtain the trust values estimated by the M vehicles and then calculate the aggregated trust value. We then compared the aggregated trust to the threshold to decide which hypothesis is true. The simulated P_{FA} values are obtained by averaging over 10^4 Monte Carlo simulations. To obtain the simulated P_{MD} values, we setup the network using hypothesis H_1 and follow the above approach. We further set $\delta_{1j}^d = \delta\ \forall j$. The results obtained for different values of μ are plotted in Figure 2. From Figure 2(a), it can be seen that as the value of δ increases, the performance of our detection system improves. We also see from Figure 2(b) that the performance improves as M increases.

Consider a situation where twenty vehicles reported trust values for an authentic RSU R_1. Out of them, suppose, M' vehicles are reporting malicious feedback. To evaluate the performance of the detection system in this situation, we have measured the false alarm probability for different values of M'. The false alarm probability is obtained similar to as described before. The only difference is that the malicious M' vehicles report false feedback as mentioned in Section V. The results for different values of μ_d are plotted in Figure 3. It can be seen from Figure 3 that the false alarm probability increases with increasing number of devices and/or increasing δ_v. However, the false alarm probability still remains close to zero unless the number of malicious vehicles exceeds the number of authentic vehicles and/or the value of δ_v is large.

Consider a situation where twenty vehicles reported trust values for ten authentic RSUs. Out of the twenty vehicles M' of them have transmitted malicious feedback, i.e. reported

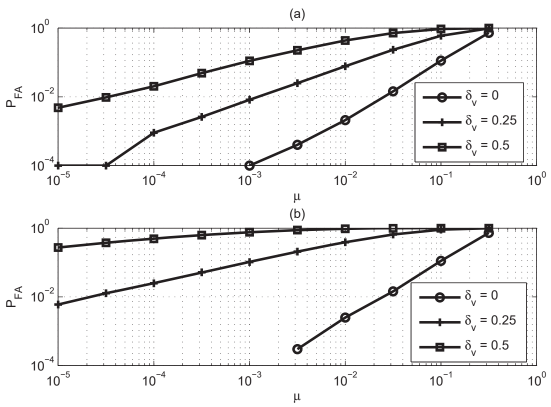

Fig. 3: Performance in the presence of malicious vehicles when (a) $M' = 2$ (b) $M' = 4$.

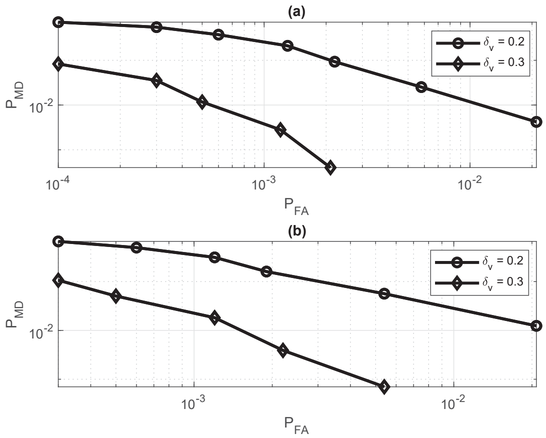

Fig. 4: Malicious vehicle identification when (a) $M' = 2$ (b) $M' = 4$.

false trust values about the ten RSUs. We use the false alarm and missed detection probabilities to characterize the performance of the Gaussian Kernel (GK) based similarity metric. To obtain the P_{FA}, in every iteration using simulations, we obtain the trust values estimated by the twenty vehicles and then calculate the aggregate trust value. The malicious M' vehicles generate feedback as mentioned in Section V. We then calculate the similarity of an authentic vehicle using (27) and then compare it with a pre-defined threshold to decide if the vehicle is authentic or malicious. The simulated P_{FA} values are obtained by averaging over 10^5 Monte Carlo simulations. The simulated P_{MD} values are obtained using a similar approach with the only difference being that the similarity of a malicious vehicle is obtained. It can be seen from Figure 4 that the similarity measure obtained in (27) achieves detection probability almost equal to one.

VII. Conclusion and Future Work

In this paper, we proposed a trust based IDS to detect the presence of a malicious RSU in vehicular networks. The detection uses the packet drop probability of the packets received as a means to obtain the trust value. To classify the RSU as malicious or authentic, the calculated trust value is compared against a threshold. In addition, we considered a situation where there could be vehicles that report false trust values about the RSUs. To identify such vehicles, we proposed to use a Gaussian kernel based similarity metric and then compared it against a threshold. The results demonstrate that the malicious RSUs and the malicious vehicles can be identified with high reliability.

An interesting future work is to detect an adversary (who has compromised an RSU) deliberately manipulating the higher layer payloads of the packets. Since such attacks cannot be correlated with the wireless channel quality, the detection system presented in this paper cannot be used to detect the attacker. Another interesting future work is to detect malicious vehicles transmitting false feedback about malicious RSUs, i.e. these vehicles try to influence the IDS into classifying the malicious RSU as authentic.

References

[1] P. Schulz, M. Matthe, H. Klessig *et al.*, "Latency Critical IoT Applications in 5G: Perspective on the Design of Radio Interface and Network Architecture," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 70–78, February 2017.

[2] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing - A key technology towards 5G," *ETSI White Paper No. 11*, 2015.

[3] B. Liang, "Mobile edge computing," in *Key technologies for 5G wireless systems*, V. W. Wong, R. Schober, D. W. K. Ng, and L.-C. Wang, Eds. Cambridge university press, 2017, pp. 1397–1411.

[4] H. Hasrouny, A. E. Samhat, C. Bassil, and A. Laouiti, "VANET security challenges and solutions: A survey," *Vehicular Communications*, vol. 7, pp. 7–20, 2017.

[5] F. Sakiz and S. Sen, "A survey of attacks and detection mechanisms on intelligent transportation systems: VANETs and IoV," *Ad Hoc Networks*, vol. 61, pp. 33–50, 2017.

[6] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges," *Future Generation Computer Systems*, vol. 78, pp. 680–698, 2018.

[7] M. Amadeo, C. Campolo, A. Molinaro, C. Rottondi, G. Verticale *et al.*, "Securing the mobile edge through named data networking," in *Proceedings of Internet of Things (WF-IoT), 2018 IEEE 4th World Forum on*, 2018, pp. 80–85.

[8] H. Han, B. Sheng, C. C. Tan, Q. Li, and S. Lu, "A timing-based scheme for rogue AP detection," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 11, pp. 1912–1925, November 2011.

[9] B. Li, T. Chen, and G. B. Giannakis, "Secure mobile edge computing in IoT via collaborative online learning," *arXiv preprint arXiv:1805.03591*, 2018.

[10] M. Hussain and B. M. Almourad, "Trust in mobile cloud computing with lte-based deployment," in *Proceedings of Scalable Computing and Communications*. IEEE, 2014, pp. 643–648.

[11] M. K. Simon and M.-S. Alouini, *Digital Communication over Fading Channels*. John Wiley & Sons, 2005.

[12] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decision Support Systems*, vol. 43, no. 2, pp. 618–644, 2007.

[13] S. M. Kay, *Fundamentals of statistical signal processing, volume i: Estimation theory (v. 1)*. PTR Prentice-Hall, Englewood Cliffs, 1993.

[14] P. Victor, C. Cornelis, M. De Cock, and E. Herrera-Viedma, "Practical aggregation operators for gradual trust and distrust," *Fuzzy Sets and Systems*, vol. 184, no. 1, pp. 126–147, 2011.

[15] N. V. Abhishek, A. Tandon, T. J. Lim, and B. Sikdar, "Detecting Forwarding Misbehavior In Clustered IoT Networks," in *Proceedings of the 14th ACM International Symposium on QoS and Security for Wireless and Mobile Networks*. ACM, 2018, pp. 1–6.

[16] J.-P. Vert and K. Tsuda, "A primer on kernel methods," *Kernel methods in computational biology*, vol. 47, pp. 35–70, 2004.

[17] P. Series, *Propagation data and prediction methods for the planning of indoor radiocommunication systems and radio local area networks in the frequency range 900 MHz to 100 GHz*. Recommendation ITU-R, 2012.