# Real-Time Detection and Localization of Denial-of-Service Attacks in Heterogeneous Vehicular Networks

Meenu Rani Dey‡, Moumita Patra‡ and Prabhat Mishra*
‡Indian Institute of Technology Guwahati, Assam, India
*University of Florida, Gainesville, Florida, USA
{rmeenu, moumita.patra}@iitg.ac.in, prabhat@ufl.edu

***Abstract*— Vehicular communication has emerged as a powerful tool for providing a safe and comfortable driving experience for users. Long Term Evolution (LTE) supports and enhances the quality of vehicular communication due to its properties such as, high data rate, spatial reuse, and low delay. However, high mobility of vehicles introduces a wide variety of security threats, including Denial-of-Service (DoS) attacks. In this paper, we propose an effective solution for real-time detection and localization of DoS attacks in an LTE-based vehicular network with mobile network components (e.g., vehicles, femto access points, etc.). We consider malicious data transmission by vehicles in two ways – using real identification (unintentional) and using fake identification. Our attack detection technique is based on data packet counter and average packet delivery ratio which helps to efficiently detect attack faster than traditional approaches. We use triangulation method for localizing the attacker, and analyze average packet delay incurred by vehicles by modelling the system as an $M/M/m$ queue. Simulation results demonstrate that our proposed technique significantly outperforms state-of-the-art techniques.**

***Index Terms*—LTE-based vehicular network, denial-of-service, real-time detection, real-time localization**

## I. Introduction

Vehicular communication has been widely investigated by researchers for providing users a smart, safe, and comfortable driving experience. To this direction, Long-Term Evolution (LTE) has recently emerged as a technology which can support vehicular applications due to its attractive features such as, high capacity, lower delay, and spatial reuse [1], [2]. However, features such as high mobility of vehicles and constantly changing network topology pose various Quality of Service (QoS) and security-related challenges [3].

LTE over the years has evolved to become a highly-complex heterogeneous system in order to support large traffic demands of users. Entities such as, Macro Base Stations (MBSs) and Femtocells/Femto Access Points (FAPs) are used to provide the increasing coverage and capacity demands [3]. FAPs are low-cost, low-power cellular base stations usually deployed in homes/offices to provide improved coverage to nearby users. Recently, mobile FAPs have gained momentum for providing better capacity and coverage in highly mobile scenarios [2], [4]. FAPs being low-power base stations are vulnerable to a wide variety of attacks including Denial of Service (DoS) attacks [5]. Additionally, mobility of FAPs further adds challenges in the detection and localization of attacks. In this work, we have considered an LTE-based Vehicular network (LTE-Vnet) scenario with mobile FAPs. Given the high mobility of vehicles as well as FAPs, detection and localization of attacks becomes highly challenging.

**Threat Model:** Our threat model is based on attack in the data plane of the network via malicious vehicles. It assumes that malicious vehicles (attackers) can be used to launch DoS attacks. We assume that malicious vehicles will transmit packets in the uplink channel to their associated FAPs, with a higher transmission rate than legitimate vehicles' transmission rate.

LTE uses Resource Blocks (RBs) for data transmission. RB is a resource unit, defined in time and frequency domain. RBs are allocated by a FAP to its associated vehicles which want to transmit data. The amount of RBs available for communication is fixed [6]. Therefore, a malicious vehicle by transmitting data at higher rates leads to higher contention for the fixed amount of available RBs. This leads to high average packet delay, low network throughput, and large packet drops. Such DoS attacks can be carried out in two possible ways:

1) Case I: high amount of fake data transmission (unintentionally) by attacker using its own Vehicle Id [7].
2) Case II: high amount of fake data transmission by attacker using fake Vehicle Id [5].

Some works in the literature study DoS attacks in LTE-Vnet scenarios and provide possible solutions to detect the attacks [6], [8], [9]. However, these works have used a less heterogeneous scenario (two layer architecture) with only MBSs. In this work, we have considered a scenario with MBSs as well as mobile FAPs in a vehicular network environment. We study the effect of DoS attack by malicious vehicles in such a scenario and propose efficient attack detection and localization techniques for the same. We intend to detect attacks at the FAPs and then localize the attackers using only the information available at the FAPs. This helps to not only make our method suitable for real-time attack detection but also helps to have a less costly approach in terms of computation. Extensive experimental results demonstrate the effectiveness of our proposed framework.

**Contributions:** The major contributions of this work are:

- We propose a real-time DoS attack detection technique in LTE-Vnet with mobile FAPs based on Packet Delivery Ratio (PDR) and Data Packet Counter (DPC).
- We propose an efficient localization technique based on triangulation method.
- We perform extensive performance analysis of average packet delay in the given scenario by modeling the system using $M/M/m$ queuing model. Simulation results demonstrate the effectiveness of the proposed framework.

## II. Related Work

Automated vehicles and electric vehicles have gained ample interest of researchers in recent years [10]–[12]. LTE in recent years has emerged as a technology which can support a large number of vehicular applications. Its properties such as low delay, high data rate, and spatial reuse help to support such communications [1], [2]. However, features such as, high mobility of vehicles and constantly changing network topology make it vulnerable to several security attacks, including DoS attacks [3], [5]. DoS attacks in LTE can take place in many ways such as, attack in network components and in data and control channels [3], [5], [13]. Several works in the literature propose solutions for overcoming security challenges in LTE [8], [9], [13], [14]. In [5], the authors have developed a theoretical framework to explore the attack space in LTE. They have shown that the attack space can be in three dimensions– communication security services, planes of attack, and network components under attack. In [14], the authors have proposed a lightweight traffic based attack detection scheme in Voice-over-LTE network. They have used Bayesian game to model their system but their work deals with static entities and the detection of attack is done at the MBS. Ambrosin et al. [9] propose a novel method to implement a distributed DoS attack on a target mobile operator's control network. They have exploited the lack of coordination between local and remote components of the LTE network during the roaming authentication process to realize a pulse DoS using temporal lensing. However, they have not proposed any attack detection or localization technique for their scenario. Authors in [13] talk about user-targeted DoS attack in LTE network. Their attack model is based on deploying a rogue base station which targets users to perform DoS attack. However they have not addressed mobility and have not proposed any detection and localization technique. Zhu et al. [8] study security flaws in platoon of vehicles in LTE-V2X networks but they have not proposed any attack detection or localization technique.

A vast majority of the existing security research efforts in LTE networks have one of the fundamental limitations: (i) they do not deal with highly mobile scenarios like vehicles, (ii) they cannot deal with heterogeneous network entities like FAPs, or (ii) they do not propose any attack detection or localization technique for their threat models. Li et al. [6] have addressed DoS attack in cellular-V2X network where attacker maliciously reserves communication resources such that legitimate vehicles get little or no resources. They have proposed an attack detection technique which is carried out at the Mobile Edge Computing (MEC) server based on the information received from the MBS. They did not consider heterogeneity in the network. Also, performing detection at MEC server and MBS may lead to high delay which makes it unsuitable for real-time attack detection. The authors have not addressed the issue of localization of attacker. While there are promising approaches for DoS attack detection and localization in network-on-chip architectures [15], [16], they are not suitable for vehicular networks.

In our work, we have considered a heterogeneous LTE-Vnet scenario with mobile FAPs and vehicles, where DoS attack is performed by certain malicious vehicles by reserving resources and thus, forcing legitimate vehicles to use little or no resources. The presence of mobile FAPs helps in spatial reuse [2] but introduces challenges in detecting the attack as well as in localizing the attacker(s). To the best of our knowledge, detection and localization of DoS attacks in such a scenario has not been handled so far in the literature, and our work is the first attempt for a solution to the aforementioned challenges.

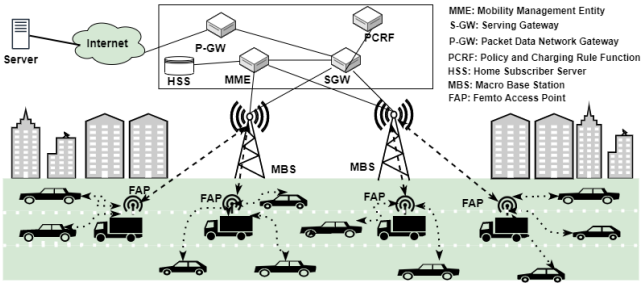


Fig. 1: System model on an LTE-based vehicular network.

## III. System Model

Our system model consists of a city scenario with multiple roads and intersections. The LTE architecture considered is given in Figure 1. Vehicles are equipped with devices which help them to communicate via cellular network. FAPs are placed on larger vehicles, such as, buses and trucks. A vehicle associates/joins with a FAP if it receives strong signal strength from the FAP. In the absence of a FAP in the neighborhood, a vehicle associates with MBS. Vehicles generate fixed size packets and the inter-arrival duration of packet generation follows an exponential distribution. Packets generated by vehicles remain in their buffer until they get a chance to transmit to a FAP or MBS. Vehicles can be either active or inactive. Active vehicles, when associated with FAPs, send packets periodically, while inactive vehicles do not send packets even after associating with a FAP.

## IV. Proposed Methodology

In this section, we propose our attack detection and localization techniques based on the threat model outlined in Section I. We perform the DoS attack detection using DPC counter and average PDR value. DPC is a counter which is initialized to the maximum number of packets that can be transmitted by a vehicle. PDR is defined as the ratio between the number of packets received to the number of generated packets.

### A. Real-Time Detection of DoS Attacks

We first consider the scenario given in Case I mentioned in the threat model (Section I), where the real identification of the attacker vehicle is available. The need here is to detect the attack. Our proposed approach for this is given below.

*1) Attack Detection for Case I:* In this case, we consider detecting the attack at the FAPs by using DPC counters. This is because all uplink packets from vehicles are transmitted via their associated FAPs. DPC counters for each vehicle, associated to a FAP, are initialized to the maximum number of packets that can be delivered by the corresponding vehicles in a given time interval. Finding the initial value of DPC counter is crucial. We show its calculation below.

Let us assume that there are $m$ RBs and $x$ vehicles are associated to a FAP. These $x$ vehicles include both legitimate

and malicious vehicles. Total time is $T$ which is divided into time slots of duration $\tau$. Let $\alpha$ be the packet generation rate. The probability of accessing any one of the $m$ RBs by a vehicle is given by

$$\delta = e^{-\alpha(\left[\frac{x-1}{m}+1\right])\tau} \quad (1)$$

Now, let us assume that $j$ out of $x$ vehicles generate packets in a given time interval. These $j$ vehicles can be selected in the following way:

$$\binom{x}{j} = \frac{x(x-1)}{j} \quad (2)$$

Let $g_t$ be the probability of generating packets by a vehicle in a given time slot, $t$. The probability of generating packets by $j$ vehicles is

$$\Gamma_j = g_t^j (1-g_t)^{x-j} \quad (3)$$

The probability that exactly $j$ of $x$ vehicles transmit packets in a given time slot is given by

$$\beta_j = j \times \Gamma_j = j \times g_t^j \left(1-g_t\right)^{x-j} \quad (4)$$

By using Equation 1 and 4, we can calculate the average number of packets transmitted per vehicle:

$$E\left[X\right] = \delta \times \beta_j \quad (5)$$

The maximum number of packets that can arrive at a FAP from a vehicle depends on the distance between the FAP and the vehicle, the Signal-to-Interference-plus-Noise Ratio (SINR) value and Reference Signal Received Power (RSRP) value. SINR is calculated as:

$$SINR = \frac{S}{N+I} \quad (6)$$

where $S$ denotes the signal strength, $N$ denotes the noise and $I$ denotes the interference in the channel. Based on SINR, the maximum channel capacity can be calculated by using Shannon's Channel Capacity theorem [2]:

$$W = B \times log_2\left(1+SINR\right) \quad (7)$$

where $B$ denotes the bandwidth of the channel. Now, by using Equation 5 and 7, the average number of packets received by a FAP, per vehicle, is calculated as:

$$E\left[r\right] = \frac{W}{E\left[X\right] \times x} \quad (8)$$

Therefore, with packet size $l$, the DPC value can be calculated as:

$$DPC = max\left(\frac{W}{l}, E\left[r\right]\right) \quad (9)$$

Algorithm 1 outlines the sequence of steps to detect and localize the attacker(s). In this algorithm, at each time step (10 ms), FAPs find their associated vehicles (line 7). At every $d$ ms, FAP checks for association with new vehicles. If present, it calculates its DPC value and initializes its packet counter $PCount$, among other values (lines 9-16). Otherwise, the uplink buffer is checked and $PCount$ is updated (lines 17-19). Based on the condition in line 20, the attack is detected and as vehicle ID is known in this case, attack is localized by flagging the corresponding vehicle (lines 20-22).

*2) Attack Detection for Case II:* In this case, we have used PDR to detect the attack. We assume that the attacker uses the ID of inactive vehicles, associated to a FAP, to launch attack and hide its own identification [5].

- Each vehicle (both legitimate and attacker) periodically calculates its own PDR with the help of acknowledgments received in each time slot.
- Vehicles then send their PDR values to FAP through control plane (the attack is happening over data plane).

**Algorithm 1: Detect & Localize the Attacker(s)**

**Input:** fap: set of FAPs
**Output:** attackerList: List of attackers

```
1  n: total number of vehicle;
2  Ubuf : Temporary uplink buffer with packet
     transmitted by each vehicle at FAP;
3  PCount:Packet count;
4  temp = 0;
5  for ((i = 1; i <= T; i++) do
6      for (f = 1; f <= fap; f++) do
7          V̄ = vehicles_associated_to_fap(f);
8          for (k = 1; k <= n; k++) do
9              if (i%d == 0) then
10                 for (v = 1; v <= V̄; v++) do
11                     if (k.VId == v.VId) then
12                         if (v.f_{i-1} ≠ v.f_i) then
13                             v.PCount = 0;;
14                             v.Attacker=False;
15                             v.Ubuf= null ;
16                             v.DPC =
                                 Calculate_DPC(v);
17                         temp = v.Ubuf.size()/l;
18                         for
                             (j = 1; j <= temp; j++)
                           do
19                             v.PCount++;
20                         if (v.PCount > v.DPC)
                           then
21                             v.Attacker = True;
22                             attackerList ← v.VId;
```

- FAP keeps track of PDR values of each vehicle associated to it and uses these values to calculate the Average PDR (APDR) of each vehicle. The objective behind calculating APDR is to avoid flagging legitimate vehicles as potential attackers.
- If APDR$< \vartheta$, a threshold [17], the vehicle is considered to be a legitimate vehicle, otherwise it is flagged as a possible attacker. The APDR value of inactive vehicles will be zero as they do not send packets. Thus, the FAP will now have two different APDR values for the same vehicle ID – one belonging to the legitimate but inactive vehicle and the other belonging to the attacker.
- If FAP gets two APDR values for the same vehicle ID, it compares both the values with the threshold and starts localizing the vehicle with the higher APDR using the method given in Section IV-B.

The attack detection is described in Algorithms 2 and 3. Algorithm 2 outlines the sequence of steps to calculate the PDR of each vehicle. Here, each vehicle associated to a FAP $f$, checks its downlink buffer (*Dbuf*) for acknowledgments received and adds the contents in a temporary list *RList* (lines $4-8$). PDR is calculated using the method given in line 9 and the corresponding value is added in the vehicle's uplink buffer *Ubuf* (line 10).

**Algorithm 2: PDR Calculation**

**Input:** Dbuf: Downlink Buffer of each vehicle with acknowledgment packets
**Output:** Ubuf: Uplink buffer of each vehicle with PDR

1 $G$ : Number of packets generated by each vehicle v;
2 $RList$ : Temporary List;
3 **for** $(i = 1; i <= T; i++)$ **do**
4  **for** $(f = 1; f <= fap; f++)$ **do**
5  $\bar{V}$ = $vehicles\_associated\_to\_fap(f)$;
6  **for** $(v = 1; v <= \bar{V}; v++)$ **do**
7  **if** $(v.Dbuf \neq null)$ **then**
8  $v.RList \leftarrow v.Dbuf$;
9  $v.pdr \leftarrow v.RList.size()/v.G$;
10  $v.Ubuf \leftarrow v.pdr$;

**Algorithm 3: Detection of DoS Attacks**

**Input:** Ubuf: uplink buffer of each vehicle with PDR
**Output:** AList: List of Attackers

1 temp=0;
2 **for** $(i = 1; i <= T; i++)$ **do**
3  **for** $(f = 1; f <= fap; f++)$ **do**
4  $\bar{V}$ = $vehicles\_associated\_to\_fap(f)$;
5  **for** $(v = 1; v <= \bar{V}; v++)$ **do**
6  **for** $(s = 1; s <= q; s++)$ **do**
7  $PList \leftarrow v.Ubuf.get(pdr)$;
8  $v.AvgPdr \leftarrow$ PList$/q$
9  **if** $(v.AvgPdr > \vartheta)$ **then**
10  $AttackerTable.put(VId, v.AvgPdr)$;
11  $EntryTable.put(VId, temp++)$;
12  **if** $(EntryTable.getvalue() > 1)$ **then**
13  **if** $(AttackerTable.getvalue() > 0)$ **then**
14  $AList$.add$(EntryTable.getkey())$

Algorithm 3 outlines the sequence of steps for detecting the attack and get a list of possible attackers. The PDR values are added to a temporary list (*Plist*) and average PDR is calculated (lines 6-8). If the average PDR is greater than a threshold value $\vartheta$ then the vehicle ID is added to a possible attackers list *AttackerTable* (line 10) and IDs of all associated vehicles (active and inactive) is added in another list, *EntryTable* (line 11). If there is any vehicle which appears more than once in the *EntryTable*, this means that it may have taken ID of an inactive vehicle. Then, its presence in the *AttackerTable* is checked. Thus, if a vehicle has taken inactive vehicle's ID and has transmitted packets more than the threshold $\vartheta$, then it is flagged as an attacker (lines 12-14).

### *B. Real-Time Localization of Attackers*

In this section, we propose our technique for localization of attackers detected in Case II. It should be noted that

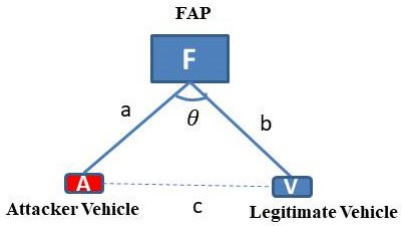


Fig. 2: Triangulation method for localization

localization in LTE-Vnet framework faces the following major challenges:

- There is no direct communication between vehicles.
- Single-hop communication takes place between vehicles and base station (FAP or MBS).
- High mobility of vehicles.

The following information is available about each vehicle: their SINR, Received Signal Strength (RSS), PDR, and number of transmitted packets. Considering the challenges and the information available about each vehicle, we propose using triangulation method for localizing the attackers. Triangulation method uses the distance of a known vehicle (legitimate vehicle) from FAP, an unknown vehicle (attacker vehicle), and the measured angle between the vehicles [18] to calculate the location of the attacker, as shown in Figure 2. As mentioned in Section III, we have considered a city scenario. Therefore, it is safe to assume that the number of vehicles is high, resulting in same relative speed between them. Although we have considered the city scenario but our work is also applicable in highway scenarios because even with high speed of vehicles, the relative speed between vehicles will remain the same. Due to low relative speed between associated vehicles and FAPs, we can use triangulation method for localization.

Let us assume that $A$ is a possible attacker obtained from Algorithm 3 at a FAP $F$, as shown in Figure 2. $F$ calculates the distance between itself to $A$ and legitimate vehicle $V$ using RSS. This distance can be calculated as [19]. Let $\gamma$ be the reference distance between transmitter $A$ and receiver $F$. The received signal power $\psi$ can be calculated as:

$$\psi\,(in\ dBm) = u - R(\gamma) \tag{10}$$

where, $R(\gamma)$ denotes reference power for $\gamma$ and $u$ as transmit power of $A$. Now, using Equation (10) the RSS value is calculated as:

$$RSS(in\ dBm) = \psi - 10h\log a \tag{11}$$

where, $h$ represents path loss exponent and depends on specific propagation environment. It measures the rate at which the RSS decreases with distance. $a$ is the distance from $A$ to $F$. Maximum RSS value $RSS_{max}$ can be obtained by computing the maximum of RSS values. Using $RSS_{max}$ value, distance $a$ is calculated as:

$$a(in\ meters) = 10^{\left(\frac{\psi - RSS_{max}}{10}\right)\times h} \tag{12}$$

The distance between $V$ and $F$ denoted as $b$ is also calculated in a similar way. After calculation of distance, the angle $\theta$ between $a$ and $b$ is calculated by FAP using following formula:

$$\theta = \tan^{-1}\left(\pm\frac{\Delta_1 - \Delta_2}{1 + \Delta_1\Delta_2}\right) \tag{13}$$

where $\Delta_1$ and $\Delta_2$ are the slope of lines $F$-$A$ and $F$-$V$. To localize the attacker, FAP needs to find the distance between $A$ and $V$. As the value of $a$, $b$ and $\theta$ are known, the distance between $A$ and $V$, which is denoted as $c$, can be calculated as follows:

$$c = \sqrt[2]{a^2 + b^2 - 2abCos\theta} \tag{14}$$

After calculating the distance, FAP calculates the common meet point using the distance $c$ and $a$. Then, it calculates the coordinates for common point and based on the coordinates it localizes the attacker.

### C. Performance Analysis

In this section, we analyze the average delay of a packet being transmitted from a vehicle to its associated base station (FAP or MBS). As mentioned in Section I, at a given time, a fixed number of RBs are available for communication and FAPs are responsible for allocating them to vehicles. Now, if we consider our attack scenarios (both Case I and Case II), we can conclude that given a fixed amount of available resources, when number of data packets increase, contention for availing the limited amount of resources will also increase, leading to high delay.

For our analysis, we trace each packet from its generation until its delivery to the associated FAP. As mentioned earlier, in the uplink, vehicles generate packets which are stored in their buffers until RB is allocated for transmission. The maximum number of vehicles that a FAP can serve is equal to the number of available RBs. On assignment of a RB, a vehicle can transfer its packets to the associated FAP. Let us assume that there are $x$ vehicles associated to a FAP. $x$ includes both legitimate and malicious vehicles. Also, there are only $m$ RBs available such that $m < G.x$, i.e., number of available RBs is less than the number of packets generated by the associated vehicles. Hence, for transmission from vehicles to FAP, vehicles have to contend for RBs. Let $E[W_v]$ be the expected waiting time incurred by packets generated at vehicles to reach FAPs. The buffers in vehicles associated to a FAP can now be modeled as an $M/M/m$ queue where $m$ RBs act as servers. Let the packet arrival rate be $\lambda$ and the service time be exponentially distributed with mean $1/\mu$. Thus, the expected waiting time for packets in vehicles is given by,

$$E[W_v] = E[P]/\lambda \tag{15}$$

where, $E[P]$ is the expected number of packets in the queue including the ones in service. $E[P]$ is given by,

$$E[P] = \frac{\rho\eta}{1-\rho} \tag{16}$$

Here, $\rho$ is the server utilization factor and $\eta$ is defined as the probability of queuing i.e, the probability that there are $m$ or more packets waiting in the queue to be served. The server utilization factor for $M/M/m$ queue is given by

$$\rho = \frac{\lambda}{m\mu} \tag{17}$$

$\eta$ is given by

$$\eta = \frac{(m\rho)^m}{m!(1-\rho)}J \tag{18}$$

$J$ represents the probability that all servers are idle and there are no packets in the system to serve.

$$J = \left[1 + \frac{(m\rho)^m}{m!(1-\rho)} + \sum_{i=1}^{m-1}\frac{(m\rho)^i}{i!}\right]^{-1} \tag{19}$$

## V. Experiments

### A. Experimental Setup

The simulation scenario consists of a city scenario with road of length 10 km with multiple lanes and intersections. The traffic is bidirectional. We are comparing our approach with [6], where the authors have considered only MBS and detection takes place at MEC server. To simulate our scenario, We have used a discrete event simulator based on Java. We have used Simulation of Urban MObility (SUMO) to generate the vehicular movements. The simulation results have been averaged over 90-100 runs. Parameters considered for simulation is given in Table I.

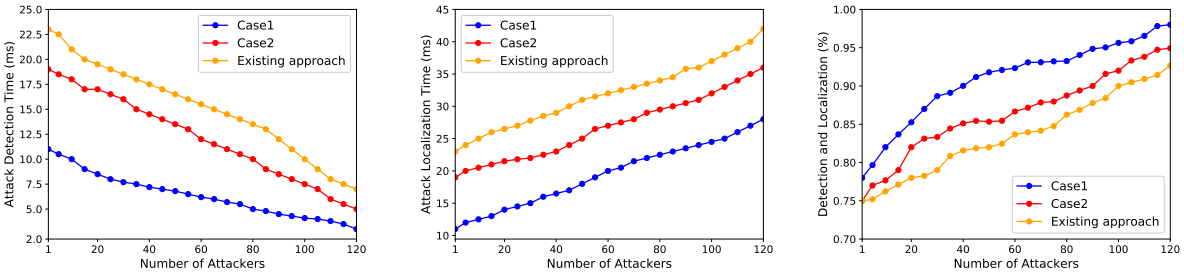
| Parameter | Value |
|---|---|
| Number of Vehicles | 300 |
| Number of FAPs | 50 |
| Packet Size | 160 bytes |
| Packet generation rate (legitimate vehicle) | 1 Packet/5ms |
| Packet generation rate (attacker) | 1 Packet/1ms |
| MBS Transmission Range | 10 km |
| FAP Transmission Range | 50 m |
| MBS Transmission Power | 43 dBm |
| FAP Transmission Power | 23 dBm |
| Vehicle Speed | 30-80 Km/hr |
| Path Loss Coefficient | FAP:3.5<br>MBS:2.5 |

TABLE I: Parameters used in simulation [2], [5]

### B. Simulation Results

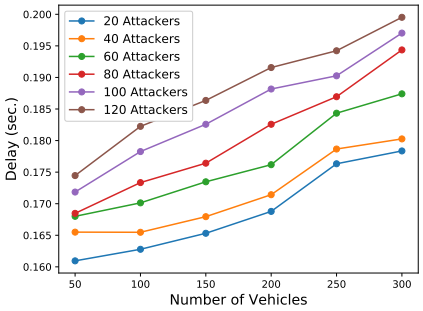
Figure 3 depicts the comparison of our approach with the existing approach with respect to detection and localization time and percentage. In the existing approach, the attack detection is done at MBS and localization of attacker is done at MEC server [6]. This takes more time and makes detection more complex. Figure 3a represents the attack detection time for Case I, Case II and existing approach. As expected, with the increase in number of attackers, the attack detection time decreases. We can see that the detection time taken in Case I is much less than that of Case II and existing approach. This is because, in Case I the detection is done at FAPs by calculating DPC values, whereas, in Case II, PDR calculated at each vehicle is transmitted to FAP which uses it for detection of attack. In the existing approach, authors have used MBS to perform detection. As the MBS is far from the vehicles and traffic load at MBS is higher than FAP, it takes more time to detect than Case I and Case II.

Figure 3b represents the time for localizing the attacker. It can be seen that Case I takes less time for localizing than both Case II and existing approach. This is because, in Case I, FAP localizes the attacker using its vehicle Id, whereas, in Case II, FAP uses triangulation method to localize the attacker, and in the existing approach neural network based method is used for localization. Figure 3c represents the variation in the percentage of attackers detected and localized with the number of attackers. As the vehicle Id is known in Case I, the attacker is easily

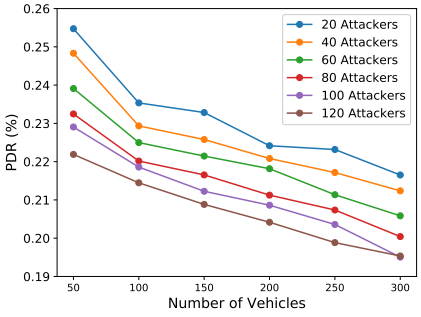



(a) Detection time with attackers (b) Localization time with attackers (c) Percentage of detection & localization

Fig. 3: Comparison of proposed technique with existing approach




(a) Impact on average packet delay




(b) Impact on PDR percentage

Fig. 4: Impact of DoS attacks with multiple attackers

detected and localized. Hence, it performs better than Case II and existing approach. Figure 4a represents the variation in average delay with increase in number of vehicles in the scenario. Vehicles include both legitimate and malicious vehicles. As expected, with increase in number of vehicles, the average delay increases. This is because, more number of vehicles will contend for the fixed amount of available resources. With the inclusion of malicious vehicles, more packets will be generated, which will eat up more resources resulting in higher contention, leading to higher delay. As we can see, higher delay can be observed when the number of malicious vehicles is more. Similarly, in Figure 4b, it can be observed that when the number of attackers increases, the average PDR percentage decreases. This is because, with more attackers, contention for available RBs increases, leading to lower PDR.

## VI. Conclusion

In this paper, we have developed an efficient framework for real-time detection and localization of Denial-of-Service (DoS) attacks in LTE-based vehicular networks. We have explored DoS attack detection using average packet delivery ratio as well as data packet counter values. We have utilized triangulation method to localize the attacker which uses fake vehicle identification to perform the attack. We have also performed detailed analysis of average packet delay using M/M/m queuing model. Our experimental results demonstrate that our approach can significantly outperform state-of-the-art methods. In future work, we plan to explore various mitigation techniques.

## Acknowledgements

This research work was supported in part by the Department of Science and Technology (DST), Government of India vide project grant $ECR/2018/000917$. This work was also partially supported by the US National Science Foundation (NSF) grant SaTC-1936040.

## References

[1] G. Araniti *et al.*, "LTE for vehicular networking: a survey," *IEEE Communications Magazine*, vol. 51, no. 5, pp. 148–157, 2013.
[2] M. Patra *et al.*, "Improving delay and energy efficiency of vehicular networks using mobile femto access points," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 1496–1505, 2017.
[3] R. Piqueras Jover, *Security attacks against the availability of LTE mobility networks: Overview and research directions.* WPMC, 2013.
[4] M. Tayyab *et al.*, *A survey on handover management: From LTE to NR.* IEEE Access, 2019, vol. 7.
[5] S. Bhattarai *et al.*, *On simulation studies of cyber attacks against LTE networks.* ICCCN, 2014.
[6] Y. Li *et al.*, *An MEC-based DoS attack detection mechanism for C-V2X Networks.* GLOBECOM, 2018.
[7] S. Mukherjee *et al.*, *Practical DoS Attacks on Embedded Networks in Commercial Vehicles.* Springer International Publishing, 2016.
[8] M. Zhu *et al.*, *Security Analysis of LTE-V2X and A Platooning Case Study.* IEEE INFOCOM, 2020.
[9] M. Ambrosin *et al.*, *A roaming-based denial of service attack on LTE networks: poster.* Association for Computing Machinery, 2017.
[10] G. Velikic, *Intelligent Cars: Are We There Yet?* IEEE Consumer Electronics Magazine, 2020, vol. 9.4.
[11] D. Baek *et al.*, "Build your own EV: A rapid energy-aware synthesis of electric vehicles." *IEEE Design Test*, pp. 40–47, 2018.
[12] H. Thapliyal *et al.*, *Emerging Paradigms in Vehicular Cybersecurity.* IEEE Consumer Electronics Magazine, 2019, vol. 8, no. 6.
[13] R. Ghannam *et al.*, *User-targeted Denial-of-Service Attacks in LTE Mobile Networks.* WiMob. IEEE., 2018.
[14] N. Ruan *et al.*, *A Traffic Based Lightweight Attack Detection Scheme for VoLTE.* IEEE GLOBECOM, 2016.
[15] S. Charles, Y. Lyu, and P. Mishra, "Real-time detection and localization of DoS attacks in NoC based SoCs," pp. 1160–1165, 2019.
[16] S. Charles, Y. Lyu, and P. Mishra, "Real-time detection and localization of distributed DoS attacks in NoC based SoCs," *IEEE TCAD*, 2020.
[17] M. Lichtman *et al.*, *LTE/LTE-A jamming, spoofing, and sniffing: threat assessment and mitigation.* IEEE Communications, 2016.
[18] C. Savarese *et al.*, *Location in distributed ad-hoc wireless sensor networks.* ICASSP, 2001.
[19] "Distance measurement using RSSI method in WSN." [Online]. Available: https://www.ukessays.com/essays/computer-science/distance-measurement-using-rssi-method-8607.php?vref=1