INFRASTRUCTURE BASED
POSITION VERIFICATION TECHNIQUE FOR VANETs

A DISSERTATION

Submitted in partial fulfillment of the
requirements for the award of the degree

of

MASTER OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

YASHIKA JAIN

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE -247 667 (INDIA)
JUNE, 2016

ACKNOWLEDGEMENTS

I express deep gratitude for enthusiasm and valuable suggestions that I got from my guide Prof. Manoj Misra, Department of Computer Science & Engineering, Indian Institue of Technology Roorkee, for successful completion of this project. His integrity and commitment has inspire me.

I heartily thank professors and technical staff of Department of Electronics and Communication Engineering, IIT Roorkee, for helping me in exploring the possibility of carrying out the experiments in their laboratory.

I owe my profound gratitude to participants from stackoverflow, specially Dr. Christoph Sommer for guiding me in completing this project.

Finally, I would like to acknowledge with gratitude, the support and love of my family and friends. They all kept me going.

Yashika Jain

ABSTRACT

VANET has been an interesting research area for years. It is a technology that uses moving cars as the nodes to create a mobile network. VANET comprises of two types of communications, V2I and V2V, accompanied by wireless communication technology, particularly IEEE 802.11p. This technology has been developed to improve road safety and to make road traffic more manageable.

A lot of research has been done to make the transportation intelligent but security aspect of VANETs got less concern. As in VANET human lives are directly involved thus it becomes necessary to pay close attention to the VANET security issues. One of the most important security issues is related to the forged position of a vehicle at any time in the vehicular network. There are various applications whose working depends upon the accuracy of position information. Thus, addressing this issue is of prime importance. In this thesis, an infrastructure based algorithm has been proposed to address position faking issue in VANETs. Its performance has been compared with one of existing infrastructure-less algorithm.

Keywords: Vehicular network, security, position forging, infrastructure-based.

.

Table of Contents

1. INTRODUCTION....1

1.1 General....1

1.1.1 Vehicular Ad Hoc Networks....1

1.1.2 VANET Security....2

1.1.3 Position Forging Attack....3

1.1.4 RADAR....3

1.1.5 Dedicated Short Range Communication (DSRC)....3

1.2 Motivation....4

1.3 Objective....4

1.4 Organization of Thesis....5

2. THEORETICAL BACKGROUND AND LITERATURE REVIEW....6

2.1 Theoretical Background....6

2.1.1 Dedicated Short Range Communication (DSRC)....7

2.1.2 Wireless Access in Vehicular Environment (WAVE)....8

2.2 Related Work....9

3. PROBLEM FORMULATION AND APPROACH USED....14

3.1 Problem Definition....14

3.2 Description....14

3.2.1 System Model....14

3.2.2 Misbehaviour Model....15

3.3 Assumptions & Dependencies....15

3.4 Methodology....15

4. PROPOSED WORK....16

4.1 Introduction....16

4.2 Description of Algorithm....16

4.2.1 Algorithm used by RSUs....17

4.2.2 Algorithm used by vehicles....18

4.2.3 Beacon Structure....18

4.3 Expected Outcome....18

5. SIMULATION TOOLS AND EXPERIMENTAL SETUP..19

5.1 Experimental Setup and Implementation..22

5.1.1 Traffic Generation...22

5.1.2 Network Simulation...23

6. RESULTS AND ANALYSIS...26

7. CONCLUSION AND FUTURE WORK..29

7.1 Conclusion...29

7.2 Future Work...29

REFERENCES...31

List of Tables

5.1 Simulation Parameters..25

List of Figures

1.1 Vehicular ad hoc network..2

2.1 DSRC frequency allocation in US...7

2.2 WAVE architecture...8

2.3 Location security mechanisms..12

4.1 Beacon Structure..18

5.1 Model Structure in OMNeT++...20

5.2 Veins Architecture..22

5.3 Road Network Topology..23

5.4 Vehicles' Movement...23

6.1 Comparative analysis of detection rate..26

6.2 Comparative analysis of false positive rate...27

CHAPTER 1

INTRODUCTION

The enhancement in the fields of hardware, software and wireless communication enabled the evolution of a completely new and revolutionary field of **V**ehicular **A**d-hoc **NET**works (VANETs). In the recent years, this field has drawn attention of various researchers and academic institutions. It has unique capability to provide on-road safety related services. VANET technology dictates two types of communications, between vehicles (V2V) and between vehicles and infrastructure (V2I). But like ebb and flow goes hand in hand, this technology along with various advantages, bring some security and privacy issues with it. VANET is the descendent of a well-known MANET (Mobile Ad-hoc NETwork) technology. Along with various advantages of parent technology, VANET inherits several security issues too. But due to its fast changing nature, there are certain other issues which are particular to VANET only, like, node impersonation attack, sybil attack, position faking attack etc.

A lot of VANET services use position information. Therefore, it becomes necessary to check its accuracy. The only way to prevent position forging attacks is to verify the broadcasted position information. Since past few years various techniques have been proposed for position verification. Few of them take infrastructure support while few others do not rely on any kind of special hardware or infrastructure.

In this thesis, an attempt has been made to detect the correctness of the position that is broadcasted by vehicles in periodic beacon packets by checking vehicle's presence at the published location, such that the number of message exchanges required are less and percentage of false alarm rates are low along with high detection rate. The suggested "Infrastructure Based Position Verification Technique (IB-PVT)" method uses RADAR to detect position of the vehicles.

1.1 General

1.1.1 Vehicular Ad Hoc Networks

Vehicular-networks represent a class of recently accepted category in wireless networks. Advances in wireless technologies and automobile industries led to the evolution of a wireless network category that causes a set of vehicles to establish a network in ad hoc manner. Thus such networks are also known as Vehicular Ad Hoc Networks (VANETs). VANETs enable the nearby vehicles to communicate. It also facilitates the communication between a vehicle and a road-side unit (RSU), as shown in the following figure 1.1.

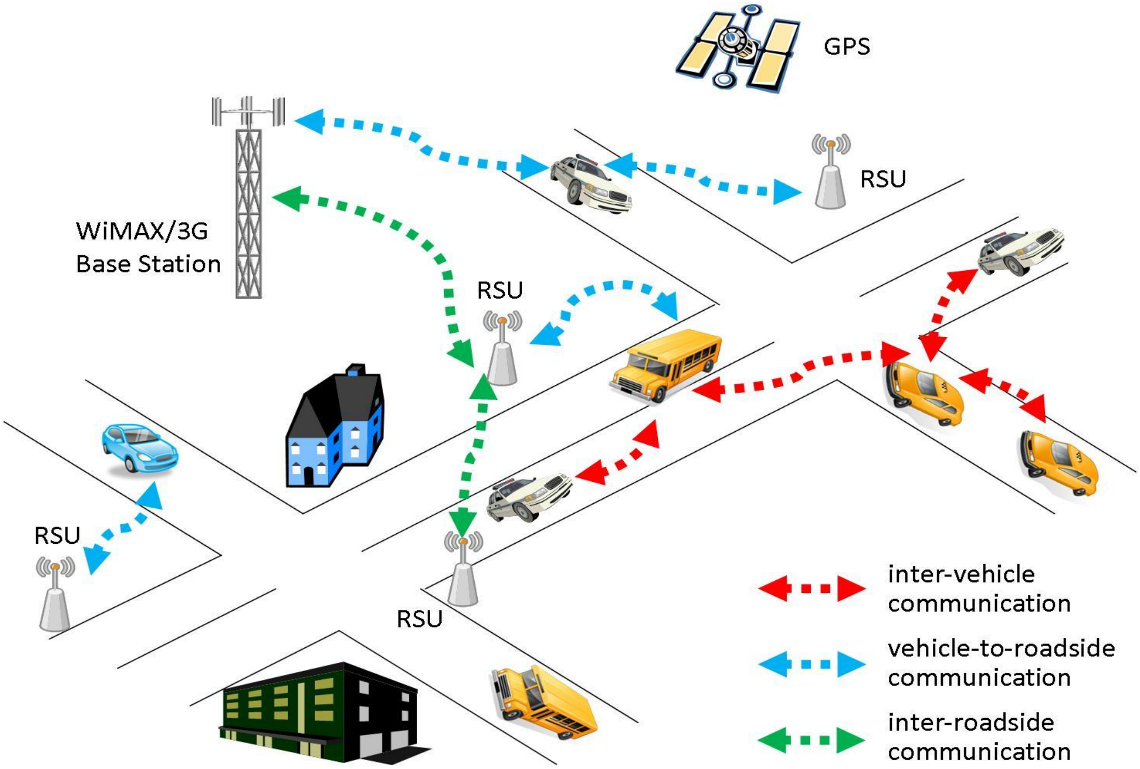
GPS
RSU
WiMAX/3G
Base Station
RSU
RSU
RSU
inter-vehicle communication
vehicle-to-roadside communication
inter-roadside communication

Figure 1.1: Vehicular Ad hoc Network [20]

Vehicular networks have attracted a great deal of attention from research community which is trying to uncover the patterns in which the vehicular communications could help in ensuring road safety and to handle traffic congestions in an intelligent manner. In the past few years researchers have made a significant growth in developing an Intelligent Transportation System (ITS).

1.1.2 VANET Security

VANET security is crucial as it directly involves human lives unlike traditional networks. The information shared among vehicles is of prime importance as depending on this information the drivers of other vehicles decide their next move. Hence, deletion or modification of such information cannot be afforded [2]. Besides communicating traffic related information, the privacy of vehicle and driver must also be observed. In order to ensure privacy of drivers, there should be a mechanism that hides the actual identity of the driver without altering the functionality of the network. Moreover, message exchange should be done in timely manner because delay in messages may bring catastrophic results such a collision of vehicles.

The deployment of a comprehensive security system for VANET is quite challenging in practice. Moreover, the nature of vehicular network is highly dynamic with frequent arrivals and departures of vehicles as well as short connection durations.

1.1.3 Position Forging Attack

Majority of the VANET applications are dependent on correct position information of a vehicle. Be it routing or notification of certain event, position plays a major role. In order to decide on the upcoming action based on the position, its correctness must be ensured. But there might be some malicious node present in the network which forges its position as well as the identity (sybil attack). This way the malicious node will be able to mimic an event that doesn't even exists. [1] Attacker might assure other vehicles of certain event like collision, in a certain pathway so that all the subsequent vehicles divert their path and the attacker gets complete path way to rash drive. Thus, for proper working of the network it is very crucial to ensure that the position information is true.

1.1.4 RADAR

RADAR is an acronym for **RA**dio **D**etection **A**nd **R**anging (RADAR). It is a system that can be used for detecting the presence, direction, speed and distance of a certain object. This technique make use of radio waves for detecting an object's presence. This is done by measuring the time for the echo of a radio wave and the direction from which it returns.

1.1.5 Dedicated Short Range Communication (DSRC)

"*DSRC (**D**edicated **S**hort **R**ange **C**ommunication) is a two-way short- to medium-range wireless communications capability that permits very high data transmission critical in communications-based active safety applications. It was developed with a primary goal of enabling technologies* *that support safety applications communication between vehicle-based devices and infrastructure to reduce collisions.*

DSRC enables the most reliable, high speed vehicle-based technology for crash prevention safety applications. It provides for a broad cross-section of dedicated connectivity options for surface transportation safety. DSRC based communications serves as the basis for connected vehicle safety and mobility application integration." [26]

1.2 Motivation

In VANETs, vehicular nodes communicate with each other in order to ensure road safety and for proper management of road traffic. For the defined objective to be fulfilled, vehicles need to exchange some messages with certain relevant information encapsulated within it. One of the most important piece of information that is communicated among nodes in a vehicular network is position information. Position information is commuted between vehicles in two ways: one is in event specific packets and another in periodic beacon packets.

The correctness of such an important information is crucial for proper working of vehicular networks. Fiddling with position information may lead to catastrophic results. This necessitates to develop an algorithm that can ensure the correctness of vehicle's position with minimal message exchanges.

1.3 Objective

The main objective of this dissertation was to develop a protocol that can verify the published position information in periodic beacon messages with minimum message exchanges. The aim was also to develop a trust system for other nodes in the network to use. An effort has been put in to develop a lightweight infrastructure-based method so that high speed vehicles are able to send and receive the relevant messages before the network disperses. The purpose was to develop a protocol that has lower computational needs for detecting the correctness of position information. The aim was also to inform every vehicle regarding behaviour of every other vehicle that it might encounter.

The protocol is expected to have lower false positive rate and higher detection rate.

1.4 Organization of Thesis

This report is organized in 7 chapters covering description of the background study of the problem along with proposed work.

Chapter 2 discusses the literature survey done so as to get state-of-the-art knowledge. It also elaborates few necessary concepts for proper understanding of the proposed scheme in dissertation.

Chapter 3 formulates the problem. It discusses assumptions and methodology used to address the defined problem.

In Chapter 4, proposed work is mentioned which is the solution for the problem discussed in previous chapter.

Chapter 5 contains the study of simulation tools and implementation details.

Chapter 6 includes results and comparative analysis of proposed algorithm with one of the existing algorithms.

Chapter 7 concludes the report and describes the work that can be done in future.

CHAPTER 2

THEORETICAL BACKGROUND AND LITERATURE REVIEW

2.1 Theoretical Background

Since past two decades continuous efforts are being made to make the vehicles smarter and driving experience securer and gratifying. In the same run, concept of -Vehicular Ad-Hoc Network (VANET) was innovated. The main ground for the conception of VANET was to inform fellow drivers about the road conditions, congestions, delays and other similar events related to traffic. The necessity for such information to flow was sensed based on the statistics of US-DOT (US Department of Transportation). According to the statistics, in a year, the congested highways incur a cost of more than $75 billion in terms of worker's productivity loss and nearly 8.4 billion gallons of fuel [19]. It was reasoned that if the drivers are given notification regarding traffic events sufficiently before, then they could make appropriate decisions regarding adopting alternate route; this would improve traffic safety simultaneously, it may contribute in saving time and fuel.

Existing hardware based technologies like inductive loop detectors, installed cameras are hugely expensive and equally not as effective. Some cost effective, functionally effective and more authentic method was required for incident reporting and traffic monitoring. This proved to be the stepping stone for VANETs. VANET utilize two types of communications Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I). To support communications between vehicles, the US FCC (Federal Communications Commission) has reserved 75MHz of spectrum in the 5.850 to 5.925 GHz band for DSRC.

The framework that includes VANETs and cooperates with Intelligent Transportation Systems (ITS) is *"Wireless Access in Vehicular Environment" (WAVE [14])*. It defines the way in which wireless devices can communicate where the physical layer properties are constantly changing and nodes in the network are highly dynamic. This section discusses about the fundamental technologies that enable working of vehicular networks.

2.1.1 Dedicated Short Range Communication (DSRC)

DSRC[14] is a short to medium range wireless communications link that renders communication between vehicles and between vehicle and infrastructure. It is a two-way RF communications link. The set of standards established for DSRC allow communication services for various kinds of applications including public safety, parking lot payment and electronic toll collection. This spectrum is useful particularly in V2X type of communications because it is the only technology that can handle very low-latency and fast network acquisition.

In the DSRC spectrum out of allocated 75 MHz, 5 MHz is set aside as guard band. The spectrum is divided into 7 channels of 10 MHz each out of which 6 of them are service channels (SCHs) and 1 is control channel (CCH). Following figure 2.1 depicts it.

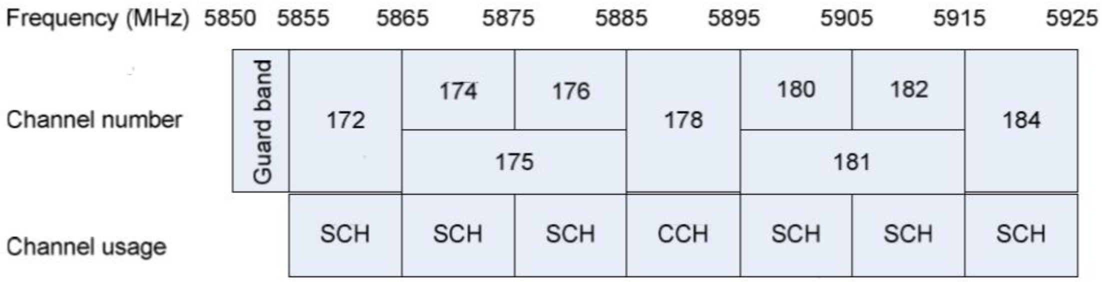

Figure 2.1: DSRC Frequency Allocation in US [14]

There are two groups of standards that support DSRC. Both have allocated different frequency bands for it: 915 MHz and 5.9 GHz. The 915 MHz DSRC standard was used even before the allocation of 5.9 GHz band. It was mainly used for commercial vehicle applications and toll collection etc. The other 5.9 GHz set of standards was majorly approved in 2006 and is undergoing through continuous revision for acceptance. It has aimed to support a large variety of applications like traffic management, collision avoidance etc. It also covers the protocols required for communication between vehicles and road-side units and between two vehicles. Currently, there are some unresolved challenges present before DSRC technology, which are being addressed by the joint efforts from academia, industry and US government.

2.1.2 Wireless Access in Vehicular Environment (WAVE) [14, 17]

WAVE architecture includes set of protocols from IEEE 1609 and IEEE 802.11p standard. The reason behind development of this architecture was to facilitate communication between moving vehicles. This standard is based on IEEE 802.11p which is an amendment to the original IEEE 802.11 standard for Wireless LAN. IEEE 802.11p standard defines the MAC and PHY layer of the WAVE architecture (figure 2.2) . The changes in IEEE 802.11p defines a manner in which fast moving vehicles can communicate through the wireless link without any requirement to set up basic service set (BSS) [14]. Thus, there is no need for authenticating nodes before data exchange. And therefore, the communication between nodes in vehicular network can happen as soon as they come into each other's range. But as authentication and association is not handled by the lower layers, the security mechanism of data confidentiality and authentication has to be handled by higher layers in the architecture.

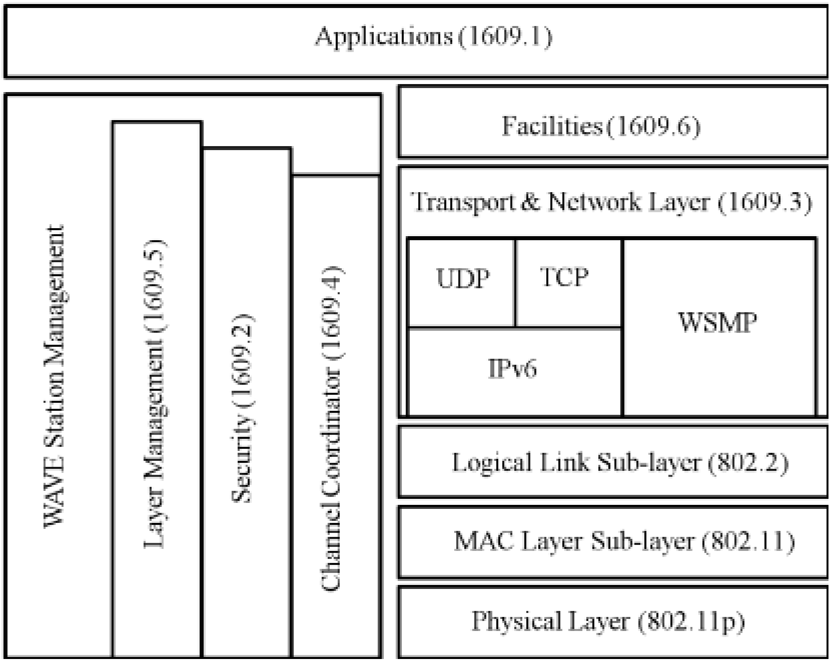
Applications (1609.1)
WAVE Station Management
Layer Management (1609.5)
Security (1609.2)
Channel Coordinator (1609.4)
Facilities (1609.6)
Transport & Network Layer (1609.3)
UDP
TCP
IPv6
WSMP
Logical Link Sub-layer (802.2)
MAC Layer Sub-layer (802.11)
Physical Layer (802.11p)

Figure 2.2: WAVE Architecture [17]

Apart from the above introduces IEEE 802.11p standard, WAVE architecture also contains IEEE 1609 standard, that governs upper layers in WAVE architecture. IEEE 1609 family takes care of security measures to be taken in order to secure the communication in vehicular environment. It

describes the services that are required for multi-channel communication. It also defines the physical aspect of communication and allows two different options for V2X communication, one is through WAVE short message and another through IPv6. Apart from these, it defines the very architecture and management structure too.
This set of standards can be used by traffic and automotive engineers who are responsible for designing, implementing and testing WAVE enabled devices.

The components of IEEE 1609 family of standard [21] are mentioned along with their corresponding status in the following points-

- **IEEE P1609.0:** *"Draft Standard for WAVE - Architecture"*
- **IEEE 1609.1:** *"Trial Use Standard for WAVE - Resource Manager"*
- **IEEE 1609.2:** *"Trial Use Standard for WAVE - Security Services for Applications and Management Messages"*
- **IEEE 1609.3:** *"Trial Use Standard for WAVE - Networking Services"*
- **IEEE 1609.4:** *"Trial Use Standard for WAVE - Multi-Channel Operations"*
- **IEEE P1609.11:** *"Over-the-Air Data Exchange Protocol for Intelligent Transportation Systems (ITS)"*

The interfaces, messages and architecture that are defined in IEEE 1609 set of standards support secure communication between vehicles and between vehicle and road-side infrastructure. These standards along with other supporting standards can be used for offering on-road safety and management services so as to reduce fatalities and other losses.

2.2 Related Work

[4] Yan G. et al. proposed a basic architecture for detecting position of a vehicle on the move. This architecture includes two components: eye-devices and ear-devices. Eye-devices are detectors like, radar, infrared sensors and camera device that act as virtual eye to "see" other vehicles. And ear-devices are meant to "hear" from other vehicles. These devices are wireless transceivers. The eye device(s) collects some data related to vehicle's position and its relative velocity. And with the ear device vehicle receives the GPS location of other vehicles. If due to

some barrier the eye device(s) cannot collect the data, then request may be sent to oncoming traffic, with very less probability of it being an attacker.

The data is received in the following manner:

$$Vr = Vl + Vre$$

where,

Vr is the detected velocity

Vl is the local velocity

Vre is the relative velocity calculated using eye device

Similarly, following is the position formula:

$$Xr = Xl + Xre$$

$$Yr = Yl + yre$$

where,

$Xr,$ Yr are coordinates collected from radar

Xl, Yl are local coordinates

Xre, Yre are relative value of xy coordinates obtained from Doppler Effect

On the data collected from eye and ear devices, cosine similarity function is applied in order to reach to an agreed-location.

[5, 6] J. Grover et al. have analyzed the effect of various forms of position forging attacks on average vehicle speed, number of collisions and percentage of packets delivered. Proposed protocol do not essentially approximate the exact position of a vehicle rather the main focus is to detect an attack itself. Different kinds of position forging attacks are detected in distinct manner. Like, a check is applied on the consistency of position information that is broadcasted in consecutive beacon packets. Along with it speed inconsistency is observed and the time for which a vehicle remains in the range of a particular RSU. Apart from detecting forging attack using single ID, this paper also observes the compound effect of both position forging and sybil attack. This is done by calculating the similarity with information of neighbours of attacker node.

Although this infrastructure-based scheme works in a number of scenarios, but in some other it is not as effective.

[7] Leinmuller T. et al. have also studied about the effects of position faking. Apart from the local effects, the simulation results have shown a global effect on the routing performance. Also, they have analyzed its effect on percentage of messages delivered. They have described the consequences of a situation when a malicious node deliberately forwards wrong position information so that greedy forwarding strategy always choose malicious node as the next forwarder and intruder can easily look into the message forwarded or in the worst case it may change the message content. The author was able to prove through simulation results, that consequences of position faking can question incredulity of network's reliability, security and performance.

[8] Leinmuller T. in his another paper extended his previous work of analyzing the effect of falsified position information by introducing an infrastructure less method to detect the correctness of position claimed by a vehicle. The proposed scheme doesn't use any additional hardware neither it requires support from infrastructure. Rather it uses various kinds of sensors to discover malicious nodes in the network. Sensor readings are used to calculate a trust value based of which trustworthiness of a vehicle can be determined. Multiple sensors that are suggested by author use data that is delivered by routing layer, so no extra hardware is required in this case. Moreover, use of multiple sensors provide fault tolerance.

[9, 10] Yan G. et al. have provided a number of methods for location security. Additionally, the authors have also covered few methods to increase the availability of location related information by selecting routes which are stable and to maintain such routes. Various methods that are provided for location security are shown in the following figure 2.3.

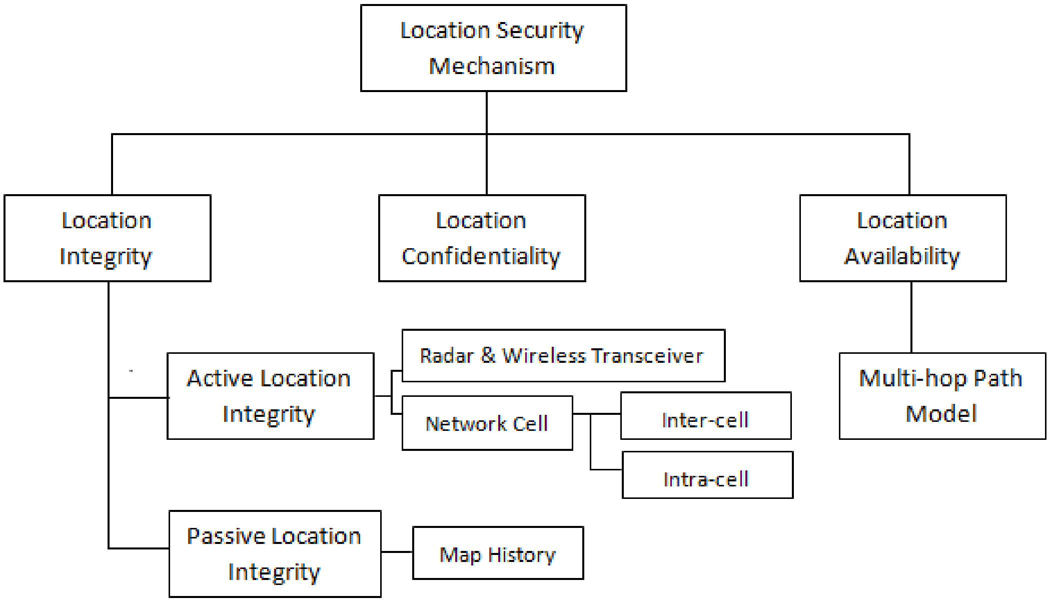

Figure 2.3: Location Security Mechanisms

[11] Abu-Elkheir M. et al. proposed a position verification approach that involves the cooperation from two-hop neighbours. In this protocol, receiver uses information gathered from direct and 2-hop neighbours to define a plausible area in which the sender should be present. Protocol uses a beacon structure where a vehicle includes its own position and the list of direct neighbours along with its judgment of their behaviour. Position of direct neighbour gives lower bound on plausibility area while position of 2-hop neighbours give upper bound on it. If the sender is found to be outside of this area, it is flagged as suspicious. However, if the sender is found to be within the plausibility area, votes received from the neighbouring nodes decide whether the sender would be flagged or not. Votes that lie into the "fresh window" are only considered.

In an sparse environment, RSS is used to approximate the distance which is then compared with the received position. If found to be noncompliant, it is flagged as suspicious. The algorithm specified by Abu-Elkheir M. et al. in their paper is used for comparing with the proposed algorithm.

[12] Fiore M. et al. have addressed the issue of position faking by providing a distributed, lightweight protocol, which relies on the information provided by the neighbouring nodes. The

protocol works in reactive manner and can be started by any node. The verifier node collects information regarding time to reach POLL and REPLY from one node to another. The time of transmission and reception are saved at each node. And once the sender reveals its identity, the collected information is fed back to the verifier. Verifier uses RF-based ToF ranging to calculate the distance between nodes and runs direct symmetry and cross symmetry tests by cross checking it with the received position. Simulation results showed that proposed protocol is effective in identifying adversarial nodes.

[13] Sastry N. et al. in their paper have suggested a method to prove whether the claimed location is in the defined region or not. They address the problem of in-region verification and not location determination. Thus, when a location 'p' is claimed to be in the region 'R', verifier 'V' accepts or rejects the claim depending on whether 'p' belongs to 'R' or not. To do this, the verifier send a nonce to the prover. As soon as prover receives this packet it 'echoes' back. This helps verifier in determining the time needed to get back the reply. So, if the prover doesn't send reply in the allowable time, the position claimed by prover is outside the region 'R'. Author asserted that the protocol doesn't use cryptography thus, it becomes less compute intensive and requires less memory. Also, this protocol doesn't require time synchronization, all it requires is that each node should be able to measure the real time with some accuracy.

CHAPTER 3

PROBLEM FORMULATION AND APPROACH USED

3.1 Problem Definition

In a vehicular wireless ad hoc network, position data is very crucial. Working of various sets of application depends on the location coordinates of the nodes in a network. But the correctness of this information is not always guaranteed. This can be due to faulty positioning device or due to deliberate alteration of the information by the user for some selfish cause.

This affects the quality of services that are provided to the users and may make driver's experiences bitter.

To address this problem, a mechanism is needed that can verify the location information for its correctness and hence assure smooth driving experience for its users.

3.2 Description

3.2.1 System Model

The system under consideration consists of smart vehicles that are equipped with On-Board Unit (OBU), Global Positioning System (GPS) device, RF antenna and some necessary sensors. All the sensors in a vehicle are considered trustworthy, though the GPS device may become faulty at any point in time. Apart from this, all the communicating units in the network are assumed to have some memory element, so that application specific relevant information can be stored. At any instant of time, the vehicles are capable of sensing their position, speed and direction. The vehicles in the network communicate with each other using DSRC communications standards and are considered to be WAVE-enabled.

It is assumed that the vehicles enter into the playground and join the traffic gradually, not all of them are present in playground before beginning of the simulation.

Apart from vehicles the network also contains RSUs such that adjacent RSUs are within range. Each of them are equipped with RADAR transceiver which would be used in verifying the location claims. Detection of moving vehicle using RADAR installed in road-side unit is already in use at some places around the globe. The communication between vehicles and RSUs is governed by DSRC standard of communication.

3.2.2 Misbehaviour Model

A node that publishes wrong position information in the periodic beacon packet is considered as misbehaving node or malicious node. This incompatibility between actual location coordinates and published location coordinates may come up either because the location sensing device is defective or because the driver has undesirable intentions. In the simulation, 10% of total nodes are considered to fake their position. Similar assumption has been used in [11]. A false position is modelled in the simulation by adding certain number to the actual location coordinates.

3.3 Assumptions & Dependencies

Following assumptions are made about the scenario:

i. All the vehicles and RSUs are WAVE-enabled,
ii. The solution is applicable in view where there are no obstacles,
iii. Every vehicle sends at least one beacon packet to every RSU,
iv. 10% of the total nodes are assumed to be malicious,
v. RSUs are assumed to have synchronized clocks,
vi. RSUs are assumed to be trustworthy.

3.4 Methodology

According to the problem definition, the problem can be divided into following tasks:

- Determine the correctness of the claimed position by RADAR device installed along-side the road.
- Save the list of vehicle IDs and corresponding trust value along with the time-stamp in the RSU.
- Flood the list of vehicle ID and corresponding unexpired trust value for the vehicles to use in determining trustworthiness of neighboring vehicles.

CHAPTER 4

PROPOSED WORK

4.1 Introduction

The proposed method, IB-PVT is infrastructure based. It makes use of additional hardware device to verify the position of a vehicle. Depending on the correctness of the information, corresponding trust values are stored against each vehicle that has claimed its location through periodic beacon packet. The protocol works in the local environment, meaning at any point the flooded position is cross checked independently by a particular RSU. So, the process is local to the vehicle and communicating RSU. Apart from the content of beacon no other information is required for completing the defined task.

4.2 Description of Algorithm

The proposed technique verifies a vehicle's presence in a certain region. The algorithm takes periodic beacon packet as an input and is triggered at RSU whenever it receives vehicle broadcasted periodic beacon. Using the information contained in the packet, vehicle's presence is verified in the appropriate region with the help of installed RADAR antenna.

Using the RADAR antenna presence of a vehicle in its range is determined. Accordingly, the trust value, 1 for being present in the checked region and 0 for being absent, against a vehicle is decided and is stored with timestamp at RSU. This process works for every received beacon. List of IDs of all the vehicles that are being encountered by an RSU, along with their corresponding trust value are flooded periodically. This information is received by vehicles on the road and it assists them to decide whether to trust a particular node or not.

Calculation of trust value at an RSU depends solely on information contained in the latest beacon broadcast. It is not related to previous broadcasts in any way. Thus, if a vehicle shows illegitimacy for a certain amount of time before being legitimate, verdict against it will be corrected before long. Thus, reducing overall false alarms rate. Unlike those algorithms that give weight-age to previous observations also, this algorithm makes use of just the latest information. Since, the trustworthiness of a node is calculated at every RSU, it's latest behaviour is always known to fellow drivers.

```
}
```

Storing trust values of all the vehicles is not necessary as following vehicles will not encounter every vehicle that has traced the same path. Thus, to reduce the storage complexity at RSU, entry corresponding an ID is removed from the table once it becomes stale. In the proposed method after 50 seconds the entry is deleted from the table.

Clocks at RSUs are considered to be synchronous. Every time a vehicle receives the verdict against another vehicle it first checks if the entry is already present in the local memory. If it's there then it compares the timestamp which is sent along in the message. Finally, the value which is latest is stored in the local memory of the vehicle.

When a vehicular node in the network receives a beacon packet, it first checks for the entry in the trust value table. If any entry matches with the received vehicle ID, the corresponding verdict is used to react in response to the message received from the neighbouring nodes. If no entry is found then vehicle simply ignores the message.

To simulate the path loss, free space propagation model is used. This model assumes that both the sender and the receiver antennas are located in an empty environment. It doesn't consider any obstacles or reflecting surfaces.

4.2.1 Algorithm used by RSUs

```
if (beacon received)
{
         extract position coordinates
        if (lower_range <= position <= upper_range)
        {
                if (Vid not present in list)
                        add_to_list (Vid, trustvalue=1, timestamp)
                else
                        update_list (Vid, trustvalue=1, timestamp)
        }
        else
        {
                if (Vid not present in list)
                        add_to_list (Vid, trustvalue=0, timestamp)
                else
                        update_list (Vid, trustvalue=, timestamp)
        }

}
```

4.2.2 Algorithm used by vehicles

```
if (list_received from RSU)
{
    for (every entry in the list)
    {
        if (Vid present in local_list)
        {
            if (ReceivedVid.timestamp < StoredVid.timestamp)
                continue;
            else
                update_list(Vid, trustvalue, timestamp)
        }
        else
            add_to_list(Vid, trustvalue, timestamp);
    }
}
```

4.2.3 Beacon Structure

<table>
<tr><td colspan="2">Vehicle_ID</td><td colspan="2">Speed</td></tr>
<tr><td>Heading</td><td colspan="2">Pos X</td><td>PosY</td></tr>
</table>

Figure 4.1: Beacon Structure

Vehicle_ID: Indicates the unique vehicle ID.

Speed: Contains vehicle's current speed in m/s.

Heading: Indicates the direction of movement.

PosX, Pos Y: Indicates the position coordinates of the vehicle.

4.3 Expected Outcome

Performance of the described protocol is measured using two metrics: false alarm rate and detection rate. *False alarm rate* or *false positive rate* is the percentage that signifies falsely accused nodes for sending incorrect position information. While *detection rate* tells the percentage of malicious nodes detected correctly. Thus, the protocol is expected to perform such that it yields high detection rate and low false positive rate by exchanging minimum number of messages.

CHAPTER 5

SIMULATION TOOLS AND EXPERIMENTAL SETUP

Referring to the facts stated earlier in this report, there are more than one standards that support DSRC. Finalizing of the principles that will guide this technology is under process. The IEEE WAVE 1609 standard is still under development. Few of its latest versions are published in the year 2016. As the guiding infrastructure for both the communicating units, RSUs and on-board unit, is under development, it seems unlikely that the actual implementation of vehicular network is possible. Thus, getting accurate results by simulating the protocol was a bit challenging. However, an attempt has been made to create as realistic scenario as possible so that the way network parameters are altered can be observed. IB-PVT is implemented and tested using OMNeT++, SUMO and Veins.

OMNeT++ [18]

OMNeT (**O**bjective **M**odular **Ne**twork **T**estbed) is a C++ simulation library, based on Eclipse IDE. It is a powerful simulation tool that follows discrete event simulation approach. As it is a modular discrete event simulator, the components of a network are defined in terms of modules. OMNeT++ supports framework approach thus it doesn't directly provide simulation components for various computer networks, rather it provides fundamental tools using which one can define their own network.

As stated above, a model or network defined in OMNeT++ consists of a number of modules. These modules function together by establishing communication among each other through message passing. There are two types of modules that can be defined in a network, namely, *compound module* and *simple module*. Compound modules act as a container for simple module as shown in the following figure 5.1. These modules do not have C++ implementation. Working of a compound module is fractioned in terms of simple modules. So, the actual active component of any network is simple module. Functionality of these modules is implemented in a C++ file while the definition and description of both the kinds of modules is given in NED (Network Description) file.

One of the most beneficial thing about modular programming approach is that a module once defined, can be reused any number of times in several other models.

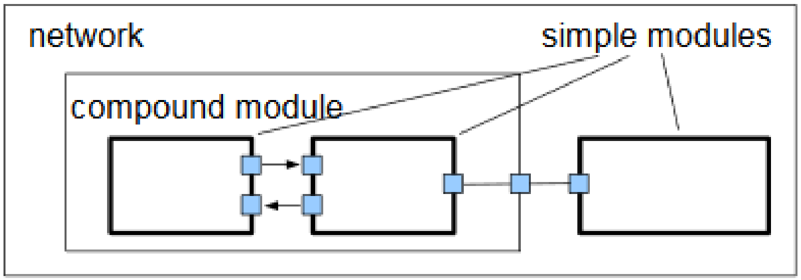

Figure 5.1: Model Structure in OMNeT++ [18]

The NED editor in OMNeT++ provides facility to define network component in both graphical as well as text form. In the graphical window, it provides a palette from which user can drag and drop the tools to form network components. While in the text mode, the components can be defined using a unique language called NED language which has java like structure for package inclusion.

In OMNeT++, modules communicate with each other by passing messages. These messages are send and received through component's output gate and input gate, respectively. Messages in OMNeT++ mimic the real world messages or packets. They may carry information within them in any data-structure. A module can send message to other module or to itself. The astounding reason for sending self-message is to help implement timer. Whenever a module receives a message, be it self-message or a message from other modules, the "local simulation time" of receiver module increases by one. Thus, to implement a timer of 10 units of time, the module will send self-message 10 times.

Before using a message, OMNeT framework must know message structure. We can write message structure in *.msg* file which will contain all the fields that a message should have. OMNeT++ provides *opp_msgc* compiler that automatically converts *.msg* files into C++ source and header files on program built. To make use of a message in a network, these message header files can be included into simple module.

One of the most important files in OMNeT++ is configuration file. As soon as the simulation starts all the NED files and configuration files are read. With every project in OMNeT++ only one configuration file is associated. This files lets the user configure the settings in which the simulation will be run. Various module parameters can be initiated through this file.

OMNeT++ also facilitates recording of scalar and vector simulation parameters in OMNeT++ output files. It also provides a mechanism to analyze these files at the end of the simulation.

SUMO [22]

Simulation of **U**rban **MO**bility (SUMO) is an open-source road traffic simulator. This simulator has the capability to handle large road networks. It allows user to define his own road-map or to make use of any network imported from google maps. It is a microscopic simulator that includes every detail required to define a road network precisely. It lets the user define, routes, number of lanes in the road, vehicle type, vehicle speed etc. By combining all the component files SUMO helps in generating network file using *netgen* command. Configuration file in sumo contains all the options that are required for starting an application. And this configuration file is run in SUMO traffic simulator.

MiXiM [23]

MiXiM is a framework built for OMNeT++ for simulating various static and mobile wireless networks. This framework plays very crucial role for simulating any wireless network. This framework contains implementation of various radio wave propagation models, path loss models like simple path loss model, log normal shadowing etc, radio transceiver power expenditure and wireless MAC protocols.

VEINS [24]

VEINS is an acronym for **VE**hicles **I**n **N**etwork **S**imulation. It is an open-source IVC (Inter-Vehicular Communication) simulation framework which consists of a road traffic model (SUMO) and a network simulator (OMNeT++). In VEINS, OMNeT++ performs network simulation while physical layer modelling is taken care by MiXiM. To facilitate IVC simulation using VEINS, both network and traffic simulators in it are bi-directionally coupled. VEINS enables online communication between these two simulators, as shown in the following figure 5.2.

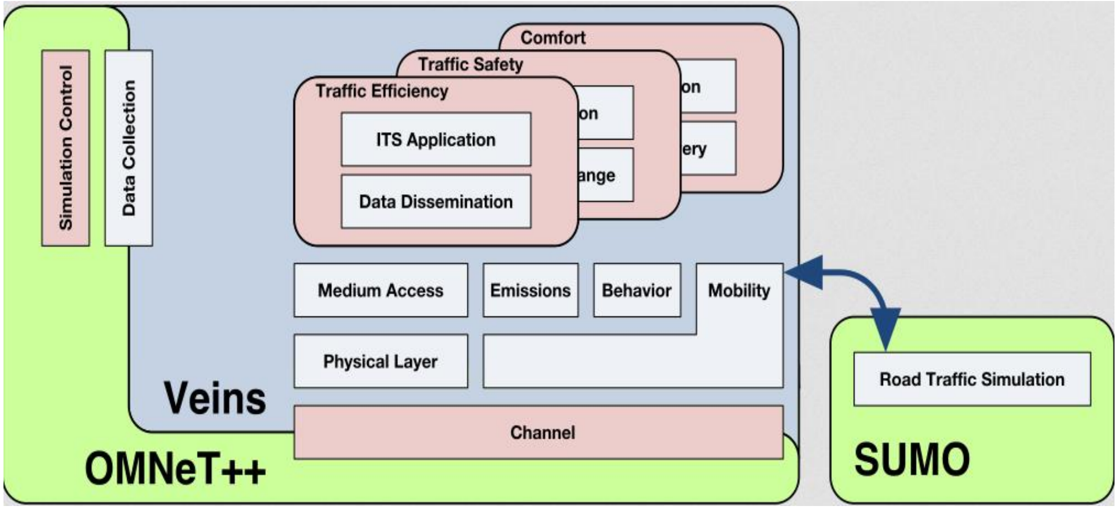

Figure 5.2: Veins Architecture [24]

Both the simulators run simultaneously and are connected using a python script "*sumo-launchd*" through local TCP socket. Message passing between them happens through this port. It helps in simulating VANET applications with realistic traffic conditions. The protocol guiding communication between network and traffic simulator has been standardized as TraCI (**Tra**ffic **C**ontrol **I**nterface). This protocol is responsible for bi-directionally coupled simulation. SUMO handles dynamics of road vehicles and transmits this information to OMNeT++. While OMNeT++ uses TraCIScenarioManager module to regularly update the change in position of every node using TraCIMobility module.

5.1 Experimental Setup and Implementation

5.1.1 Traffic Generation

In this thesis, "sumo-0.21.0" is used to generate the desired traffic. The topology that is used in this thesis is same as used in [11]. The experiment is performed with 6 different scenarios containing different number of vehicles. The reason behind it was to see the performance of the protocol under sparse, dense and semi-dense traffic conditions.

Road network is shown in the following figure 5.3-

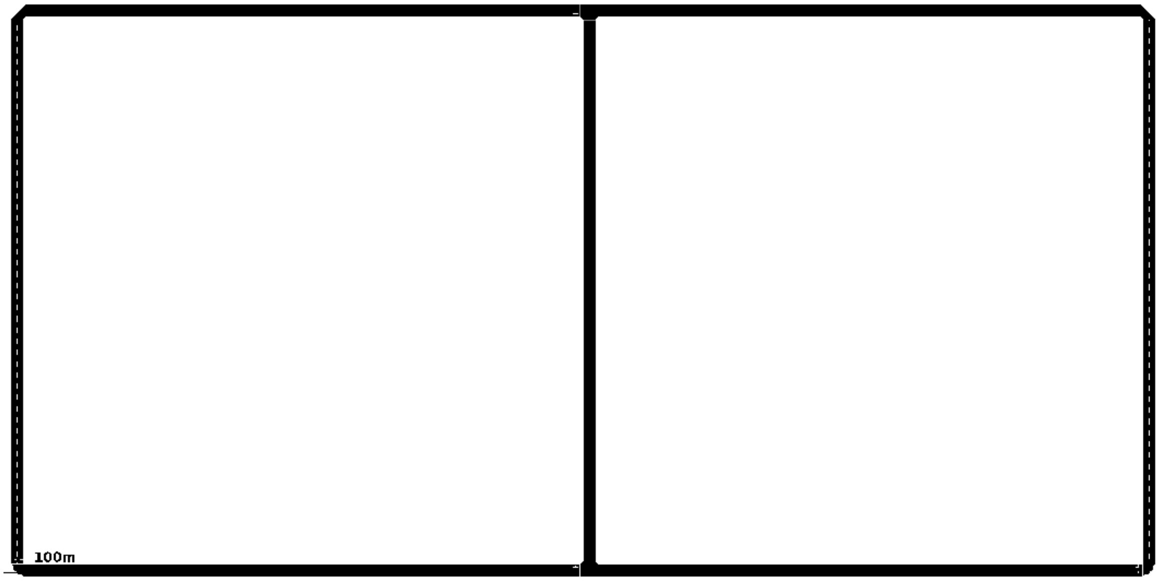

Figure 5.3: Road Network Topology

Following close-up image shows the vehicles movement (figure 5.4)-

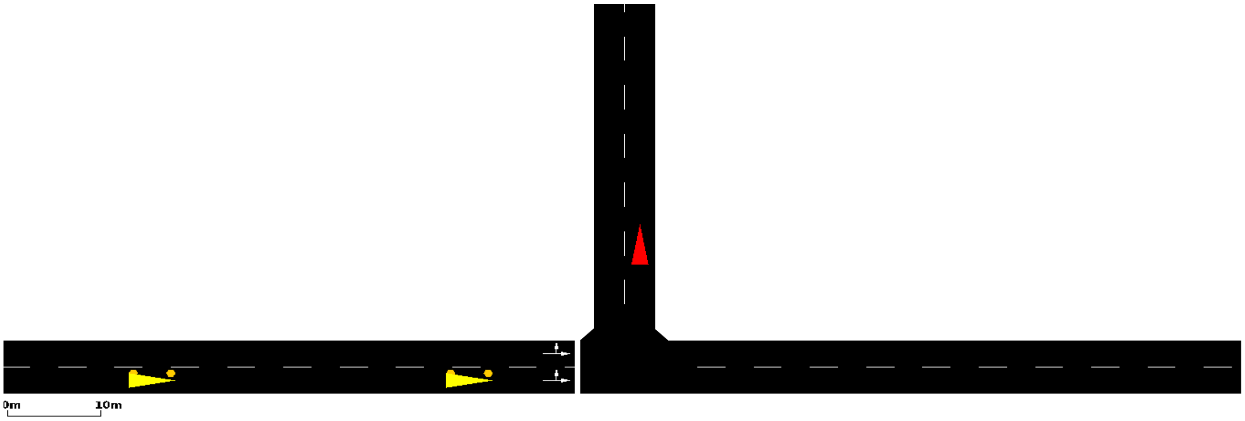

Figure 5.4: Vehicles' Movement

5.1.2 Network Simulation

For implementing the proposed application layer protocol, IB-PVT, "omnetpp-4.6" is used along with "veins-3.0".

- In the simulation scenario, 10% of the total nodes are considered to be malicious. These nodes fake their position several times during the complete simulation process [11].

Altered positions are generated by adding certain constant to the actual location coordinates. Some of the nodes are made to behave maliciously for first few seconds only. For the remaining time they are made to behave legitimately. This is done by applying a 'for loop' in the appropriate module.

- The desired beacon message is implemented by defining a *.msg* file in the project. This beacon message contains VehicleID, speed, direction, position coordinates and timestamp. C++ source and header files for the defined beacon message are generated by OMNeT++ itself using *opp_msgc* compiler.
- Apart from beacon message, a message from RSU that contains a list of unexpired entries of VehicleID and its corresponding trust value, is also used. The validity of a vehicle's trust value lasts for 50 seconds from the time it has been recorded.
- The table of vehicle's trust value along with its ID and timestamp, that is stored at every RSU and in every vehicle, is implemented using linked list. This list is forwarded after every 2 seconds.
- As for detecting the presence of a vehicle on a particular location using RADAR, we need to use a path loss model to model the signal attenuation. In our simulation, free path loss model is used.
- In order to receive the propagated signal, the resultant signal strength after all the attenuation should be more than receiver's sensitivity, only then the signal will be detected at the receiver. To implement this in the protocol, suitable receiver's sensitivity is defined.

The simulation is run for six different traffic conditions, containing 50, 100, 150, 200, 250 and 300 nodes. The performance of the protocol is recorded, in all the scenarios, in the terms of detection rate, false positive rate and total number of message exchanges.

Following table summarizes the simulation parameters used. Some of the following parameters are same as used in [11]-

Simulation Parameter	**Value**
Number of nodes	50, 100, 150, 200, 250, 300
Malicious nodes percentage	10
Max. node velocity (Kmph)	120
Number of lanes	2
Beaconing interval	6 sec
Path loss model	Free space path loss
MAC layer	IEEE 802.11p
Simulation time	1000s

Table 5.1: Simulation Parameters

CHAPTER 6

RESULTS AND ANALYSIS

This aim of above discussed protocol was to detect falsified position information with higher percentage of detection rate and to mark deserving nodes to be trustworthy thereby reducing the percentage of false positives. The purpose was also to detect position falsification and inform fellow drivers about untrustworthy nodes in minimum number of message exchanges. The proposed method was compared with the existing infrastructure-less trust based protocol described in [11]. After simulating and testing both the protocols in all different traffic scenarios, following results are obtained-

Detection Rate-

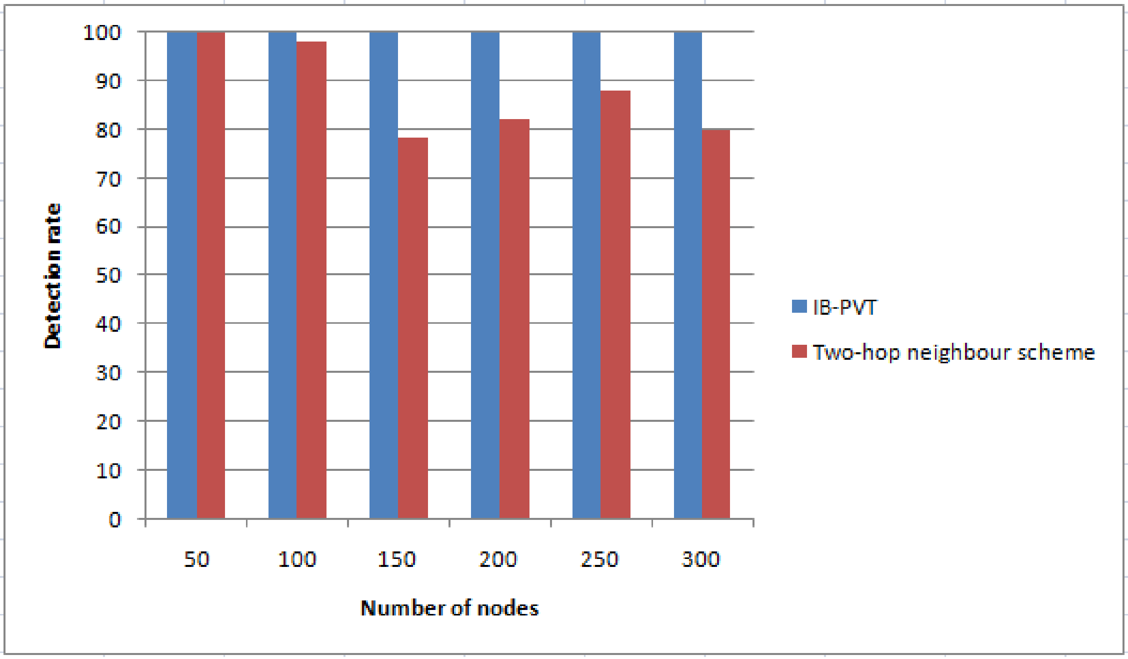

Figure 6.1: Comparative analysis of detection rate

By studying the above graph, we can say that proposed protocol has detected almost all the fraudulent nodes. While on the other hand, existing protocol has shown varied rate of detection from 78.2% to 100%. The reason for the drop in detection rate is due to the fact that the vehicles have been able to collect votes in favour of themselves by showing desirable behaviour for some time. Thus, in the case when there are more number of nodes, and published falsified position is within the two-hop connectivity, majority of votes indicate it to be legitimate node while in reality it is fraudulent, this way the detection of malicious behaviour goes unnoticed.

False Positive Rate-

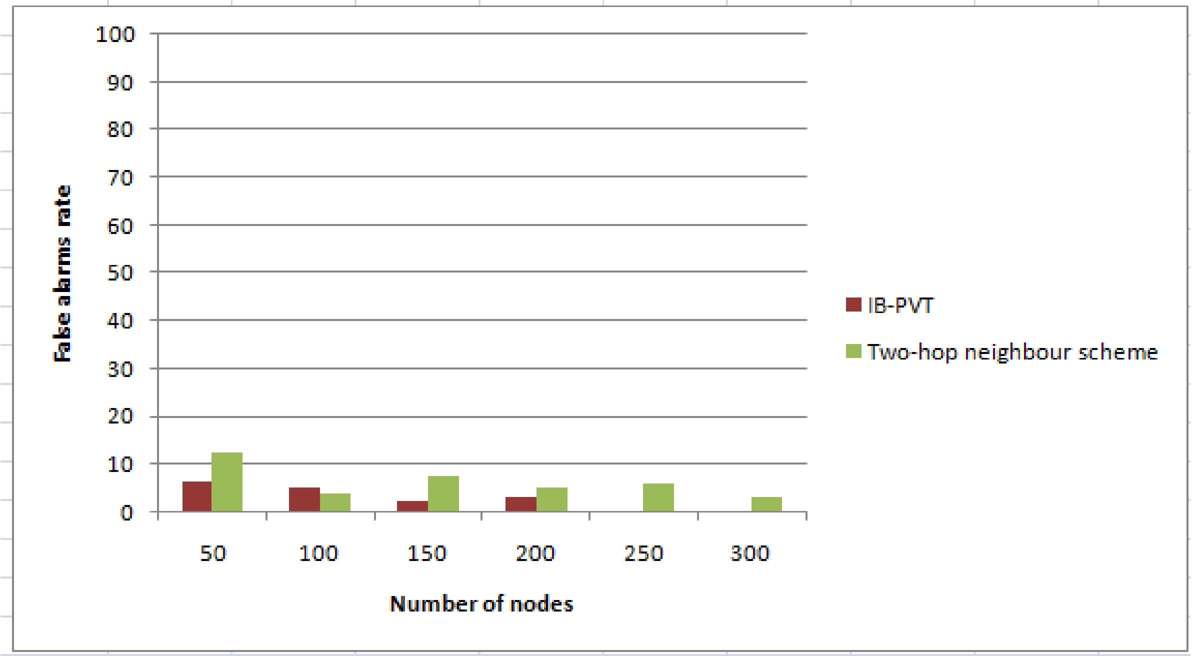

Figure 6.2: Comparative analysis of false positive rate

In the proposed algorithm, the false alarm rate varies from 0% to 6.3%. It is seen, because of the error factor in the calculations. On the other hand, in infrastructure less algorithm percentage of false alarms range from 12.4% to 3%. In the sparse environment, the false alarms are observed because of the error factor in RSS (Received Signal Strength) measurements. While in high density traffics, the false alarms are due to change in behaviour of vehicles from malicious to legitimate. As initially, the malicious node might have collected the votes against him and when its behaviour changed to legitimate, majority of voting went against him regardless of it being legitimate. This change in behaviour on single ride from malicious to non-malicious led to generation of false alarms.

Number of messages-

The only messages that are required in proposed algorithm are periodic beacon messages and the messages that are regularly flooded by RSU containing trust value of vehicles.

On the other hand, in the algorithm with which the performance of proposed algorithm is compared, apart from periodic beacon messages, few messages to determine 2-hop neighbours are also required. In addition to it, this protocol also mentions vote collection from the neighbouring nodes. Though in [11] it is mentioned that the beacon packet itself contains list of verdicts against fellow vehicles, even then the size of beacon message in this case will be more comparatively.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 Conclusion

For the vehicular ad hoc network, proposed method was designed to verify the broadcasted position information by vehicles in periodic beacon packets. The aim was to minimize the number of messages required to determine the correctness of the location so that the system can work even when vehicles are running at a very high speed. And for the reason stated above a low-computation, light-weight algorithm was required so that it can produce results quickly. The simulation results show that the described algorithm is able to detect position falsification in almost all the cases with ~100% detection rate. However, the false alarm rates are still generated by the algorithm, but the percentage of false alarms are not very high and are in satisfactory limits.

7.2 Future Work

The method that is discussed in this report assumes free space path loss model. In future, such a method could be simulated in more realistic environment, considering various other realistic parameters apart from the path loss model. Because this method is infrastructure based, there are various other factors that needs to be considered while evaluating its performance. Climatic conditions, nature of road traffic, number of lanes on the road etc. are some of the crucial factors that will affect working of hardware supported infrastructure based method. Also RADAR systems are susceptible to noise, thus, performance of RADAR supported method cannot be guaranteed in high noise areas.

While on one hand, infrastructure supported mechanism performs well in terms of detection and false alarm rates, but the cost factor cannot be neglected. The additional hardware required to ensure proper functioning of vehicular network requires more monetary investments as compared to what will be required in infrastructure less method. Moreover, the hardware based methods can work only in the presence of specific hardware. This reduces the scope of its applicability. On the other hand, infrastructure-less methods are independent of any such dedicated hardware of infrastructure.

Thus, in future an attempt should be made to develop a fused method that will have the ability to be as reliable as any infrastructure-based method and as independent and as cost-effective as any infrastructure-less method.

REFERENCES

[1] J.-P. Hubaux , S. Capkun and J. Luo "The security and privacy of smart vehicles", *IEEE Security & Privacy Magazine*, vol. 2, no. 3, pp.49 -55 2004.

[2] A. Aijaz , B. Bochow , D. Florian , A. Festag , M. Gerlach , R. Kroh and L. Tim "Attacks on inter vehicle communication systems—An analysis", *Proc. 3rd Int. Workshop Intell. Transp.*, 2006.

[3] M. Raya, J Pierre Hubaux, "The security of VANETs", Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks, 2005.

[4] Y. Gongjun, B. B. Bista, D. B. Rawat and E. F. Shaner, "General active position detectors protect VANET security", *Proc. Int. Conf. Broadband Wireless Comput., Commun. Appl.*, pp.11 -17 2011.

[5] J Grover, V Laxmi, MS Gaur, "Attack models and infrastructure supported detection mechanisms for position forging attacks in vehicular ad hoc networks", CSI transactions on ICT, Springer.

[6] Jyoti Grover, Manoj Singh Gaur, Vijay Laxmi, "Position Forging Attacks in Vehicular Ad Hoc Networks: Implementation, Impact and Detection", Wireless Communications and Mobile Computing Conference, 7th International, Conference, July 2011.

[7] T. Leinmüller, E. Schoch, F. Kargl, and C. Maihöfer, "Influence of Falsified Position Data on Geographic Ad Hoc Routing", in Proc. ESAS, 2005.

[8] T. Leinmuller , E. Schoch and F. Kargl "Position Verification Approaches for Vehicular Ad Hoc Networks", *IEEE Wireless Commun.*, vol. 13, no. 5, pp.16 -21 2006.

[9] G. Yan , S. Olariu and M. Weigle "Providing location security in vehicular ad hoc networks", *IEEE Wireless Commun.*, vol. 16, no. 6, pp.48 -55 2009.

[10] G. Yan , S. Olariua and M. C. Weigle "Providing VANET security through active position detection", *Comput. Commun.*, vol. 31, no. 12, pp.2883 -2897 2008.

[11] M. Abu-Elkheir , S.A. Hamid , H.S. Hassanein , I.M. Elhenawy and S. Elmougy "Position Verification for Vehicular Networks via Analyzing Two-Hop Neighbors Information", *Proc. IEEE 36th Conf. Local Computer Networks (LCN),* 2011.

[12] M. Fiore, C. Casetti, C.-F. Chiasserini and P. Papadimitratos, "Secure Neighbor Position Discovery in Vehicular Networks", *Proc. IEEE/IFIP 10th Ann. Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net),* June 2011.

[13] N. Sastry, U. Shankar, and D. Wagner, "Secure Verification of Location Claims", *Proc. ACM Wksp. Wireless Security*, 2003.

[14] Yunxin (Jeff) Li, "An overview of DSRC/WAVE technology", in *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering,* vol. 74, 2010.

[15] D. Kissinger, "RADAR Fundamentals", *Millimeter-Wave Receiver Concepts for 77GHz Automative Radar in Silicon-Germanium Technology,* part of the series *SpringerBriefs* in Electrical and Computer Engineering, pp 9-19, 2012.

[16] A. Wegener, M. Piorkowski, M. Raya, H. Hellbrü,ck, S. Fischer and J.-P. Hubaux, "TraCI: An Interface for Coupling Road Traffic and Network Simulators," *Proc. 11th Comm. and Networking Simulation Symp. (CNS ',08),* Apr. 2008.

[17] S. A. Mohammad, A. Rasheed, and A. Qayyum, "VANET Architectures and Protocol Stacks: A Survey," in Communication Technologies for Vehicles. Springer, 2011, pp. 95-105.

[18] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment", in Proceedings of the First International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SIMUTools 2008'), March 2008.

[19] M. Eltoweissy, S. Olariu and M. Younis, "Towards Autonomous Vehicular Clouds," *Proc. Int', Conf. Ad Hoc Networks (AdHocNets ',10),* Aug. 2010.

[20] Web reference-
http://adrianlatorre.com/projects/pfc/#/slide5

[21] Web reference-
https://www.standards.its.dot.gov/

[22] Web reference-
http://sumo.dlr.de/wiki/SUMO_User_Documentation

[23] Web reference-
http://mixim.sourceforge.net/

[24] Web reference-
http://veins.car2x.org/

[25] Web reference-
http://www.radartutorial.eu/

[26] Web reference-
http://www.its.dot.gov/factsheets/dsrc_factsheet.htm