IOSR Journal of Computer Engineering (IOSRJCE)
ISSN: 2278-0661 Volume 3, Issue 3 (July-Aug. 2012), PP 28-38
www.iosrjournals.org

VANET: Routing Protocols, Security Issues and Simulation Tools

Mushtak Y. Gadkari[1], Nitin B. Sambre[2]
[1](*Information Technology, RMCET, Ambav/Mumbai University, India)*
[2](*Electronics& Telecommunication, KIT, Kolhapur/ Shivaji University, India)*

Abstract: *Since the last few years VANET have received increased attention as the potential technology to enhance active and preventive safety on the road, as well as travel comfort. Several unexpected disastrous situations are encountered on road networks daily, many of which may lead to congestion and safety hazards. If vehicles can be provided with information about such incidents or traffic conditions in advance, the quality of driving can be improved significantly in terms of time, distance, and safety. One of the main challenges in Vehicular ad hoc network is of searching and maintaining an effective route for transporting data information. Security and privacy are indispensable in vehicular communications for successful acceptance and deployment of such a technology. The vehicular safety application should be thoroughly tested before it is deployed in a real world to use. Simulator tool has been preferred over outdoor experiment because it simple, easy and cheap. VANET requires that a traffic and network simulator should be used together to perform this test. In this paper, the author will make an attempt for identifying major issues and challenges associated with different vanet protocols, security and simulation tools.*

Keywords: *Adversaries, Attacks, Mobility generators, Protocols, VANET.*

I. INTRODUCTION

Vehicular ad hoc network (VANET) is a vehicle to vehicle (Inter-vehicle communication-IVC) and roadside to vehicle (RVC) communication system. The technology in VANET integrates WLAN/cellular and Ad Hoc networks to achieve the continuous connectivity (Fig-1). The ad hoc network is put forth with the novel objectives of providing safety and comfort related services to vehicle users [1]. Collision warning, traffic congestion alarm, lane-change warning, road blockade alarm (due to construction works etc.) are among the major safety related services addressed by VANET. In the other category of comfort related services, vehicle users are equipped with Internet and Multimedia connectivity.

The major research challenges in the area lies in design of routing protocol, data sharing, security and privacy, network formation etc. We aim here to study the efficacy of communication network in VANET on the basis of a predictable mobility model

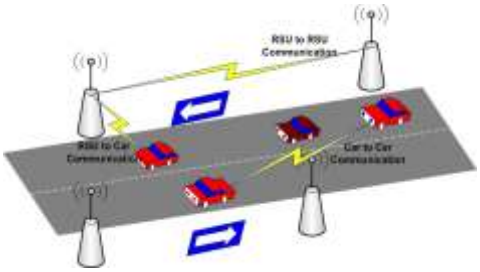

Figure 1:VANET

1.1. CHARACTERISTICS OF VEHICULAR ADHOC NETWORKS

- Do not need any infrastructure.
- Self Organized and distributed network
- High mobility nodes
- Predictable topology (using digital map)
- Critical latency requirements
- Slow migration rate
- No problem with power

1.2. VANET Applications

Public Safety

- Co-operative Collision warning [V-V]
- Intersection Collision Warning

- Approaching Emergency Vehicle
- Work Zone Warning [R-V]

Non-Public Safety

- Electronic Toll Collection
- Data Transfer
- Parking Lot Payment
- Traffic Information

II. Routing Protocol

In VANET, the routing protocols are classified into five categories: Topology based routing protocol, Position based routing protocol, Cluster based routing protocol, Geo cast routing protocol and Broadcast routing protocol. These protocols are characterized on the basis of area / application where they are most suitable.

2.1. Topology Based Routing Protocols

These routing protocols use links information that exists in the network to perform packet forwarding. They are further divided into Proactive, Reactive & Hybrid Protocols.

2.1.1. Proactive routing protocols

The proactive routing means that the routing information, like next forwarding hop is maintained in the background irrespective of communication requests. The advantage of proactive routing protocol is that there is no route discovery since the destination route is stored in the background, but the disadvantage of this protocol is that it provides low latency for real time application. The various types of proactive routing protocols are: FSR, DSDV, OLSR, CGSR, WRP, and TBRPF.

2.1.2. Reactive/Ad hoc based routing

Reactive routing opens the route only when it is necessary for a node to communicate with each other. Reactive routing consists of route discovery phase in which the query packets are flooded into the network for the path search and this phase completes when route is found. The various types of reactive routing protocols are AODV, PGB, DSR, TORA, and JARR.

2.1.3. Hybrid Protocols

The hybrid protocols are introduced to reduce the control overhead of proactive routing protocols and decrease the initial route discovery delay in reactive routing protocols. The various types of hybrid protocols are ZRP, HARP.

2.2. Position Based Routing Protocols

Position based routing consists of class of routing algorithm. They share the property of using geographic positioning information in order to select the next forwarding hops. Position based routing is broadly divided in two types: Position based greedy V2V protocols, Delay Tolerant Protocols

2.3. Cluster Based Routing Protocols

Cluster based routing is preferred in clusters. A group of nodes identifies themselves to be a part of cluster and a node is designated as cluster head will broadcast the packet to cluster. Good scalability can be provided for large networks but network delays and overhead are incurred when forming clusters in highly mobile VANET. The various Clusters based routing protocols are COIN, LORA-CBF, TIBCRPH, and CBDRP.

2.4. Geo Cast Routing Protocols

Geo cast routing is basically a location based multicast routing. Its objective is to deliver the packet from source node to all other nodes within a specified geographical region (Zone of Relevance ZOR). The various Geo cast routing protocols are IVG, DG-CASTOR and DRG.

2.5. Broadcast Based Routing Protocols

Broadcast routing is frequently used in VANET for sharing, traffic, weather and emergency, road conditions among vehicles and delivering advertisements and announcements. The various Broadcast routing protocols are BROADCOMM, UMB, V-TRADE, and DV-CAST.

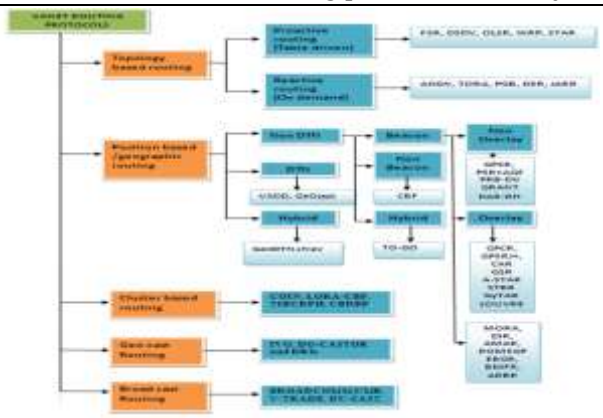

Figure 2:Routing protocols for VANET

III. Security Issues Of Vehecular Networks

VANET facing many attacks; these attacks are discussed in the following subsections:

3.1. ATTACKS AND THREATS

In this paper we are concentrating on attacks perpetrated against the message itself rather than the vehicle, as physical security.

3.1.1. Denial of Service attack

This attack happens when the attacker takes control of a vehicle's resources or jams the communication channel used by the Vehicular Network, so it prevents critical information from arriving. It also increases the danger to the driver, if it has to depend on the application's information. For instance, if a malicious wants to create a massive pile up on the highway, it can make an accident and use the DoS attack to prevent the warning from reaching to the approaching vehicles [2].

3.1.2. Message Suppression Attack

An attacker selectively dropping packets from the network, these packets may hold critical information for the receiver, the attacker suppress these packets and can use them again in other time[3]. The goal of such an attacker would be to prevent registration and insurance authorities from learning about collisions involving his vehicle and/or to avoid delivering collision reports to roadside access points [4].

3.1.3. Fabrication Attack

An attacker can make this attack by transmitting false information into the network, the information could be false or the transmitter could claim that it is somebody else. This attack includes fabricate messages, warnings, certificates, Identities [3][4].

3.1.4. Alteration Attack

This attack happens when attacker alters an existing data. It includes delaying the transmission of the information, replaying earlier transmission, or altering the actual entry of the data transmitted [3].

3.1.5. Replay Attack:

This attack happens when an attacker replay the transmission of earlier information to take advantage of the situation of the message at time of sending

3.1.6. Sybil Attack

This attack happens when an attacker creates a large number of pseudonymous, and claims or acts like it is more than a hundred vehicles, to tell other vehicles that there is jam ahead, and force them to take alternate route [3].

3.1.7. Eavesdropping is the most prominent attack over VANETs against confidentiality. To perform it, attackers can be located in a vehicle (stopped or in movement) or in a false RSU. Their goal is to illegally get access to confidential data. As confidentiality is needed in group communications, mechanisms should be established to protect such scenarios.

3.2. ADVERSARIES

3.2.1. Selfish Driver

The general idea for trust in Vehicular Network is that all vehicles must be trusted initially, these vehicles are trusted to follow the protocols specified by the application, some drivers try to maximize their profit from the network, regardless the cost for the system by taking advantage of the network resources illegally [3]. A Selfish Driver can tell other vehicles that there is congestion in the road, so they must choose an alternate route, so the road will be clear for it[4].

3.2.2. Malicious Attacker

This kind of attacker tries to cause damage via the applications available on the vehicular network. In many cases, these attackers will have specific targets, and they will have access to the resources of the network [3].

3.2.3. Pranksters

Include bored people probing for vulnerabilities and hackers seeking to reach fame via their damage [3]. For instance, a prankster can convince one vehicle to slow down, and tell the vehicle behind it to increase the speed.

3.2.4. Industrial Insiders

Industrial insiders are those who stays inside the car manufacturing company Attacks from insiders can be very harmful, and the extent to which vehicular networks are vulnerable will depend on other security design decisions.

3.2.4 Malicious Attackers

This kind of attackers deliberately attempt to cause harm via the applications on the vehicular network. Normally, these attackers have specific targets, and they have access to more resources than other attackers. They are more professional.

IV. Security Proposals Over Vanets

In recent years, there have been a plethora of contributions related to VANET security. All those previous works are based on different techniques to achieve their security goals and so to protect VANETs against the described attacks.

4.1. Security Hardware

Among the vehicle onboard equipment, there should be two hardware modules needed for security, namely the Event Data Recorder (EDR) and the Tamper-Proof Device (TPD). Whereas the EDR only provides tamper-proof storage, the TPD also possesses cryptographic processing capabilities. The EDR will be responsible for recording the vehicle's critical data, such as position, speed, time, etc., during emergency events, similar to an airplane's black box. These data will help in accident reconstruction and the attribution of liability. EDRs are already installed in many road vehicles, especially trucks. These can be extended to record also the safety messages received during critical events[4].

4.2. Identification mechanisms

Vehicular contexts have an interesting feature related to identity management. As opposed to classical computer networks, in which no central registration exists, vehicles are uniquely identified from the beginning. Indeed, this process is performed by both manufacturers and the legal authority. Manufacturers assign each vehicle a Vehicle Identification Number (VIN). On the other hand, legal authorities require vehicles to have a license plate. Both identifiers are different by nature. Whereas VINs are intended to uniquely identify manufactured vehicles, license plates are assigned to every vehicle registered in an administrative domain. Thus, VINs cannot be changed for a given vehicle, whereas license plates can change over time. Moreover, license plates are intended to be externally visible. This issue has an immediate consequence related to privacy preservation - vehicles are not completely anonymous, as visible tracking is currently possible[4].

4.3. Authentication and privacy issues

With respect to electronic identification, a natural extension of license plates called Electronic License Plate (ELP) .This credential is issued by the legal authority, allowing vehicles not only to get identified, but also to authenticate themselves. However, as this credential includes the vehicle´s real identity, it makes possible to track a vehicle. Thus, it is necessary to design a mechanism that balances authentication and privacy. Public key certificates are envisioned for this purpose. These are electronic documents that link a public key with a subject´s identity. However, using real or permanent identity would allow tracking. As opposed from that, these

credentials should not make the vehicle to be completely anonymous. Liability attribution is required by the legal authority whenever misbehavior (e.g. traffic offence, false warning) is detected. This tradeoff is called resolvable anonymity. Two different mechanisms have been proposed to satisfy this need in VANETs – identity-based cryptography and pseudonymous short-lived public key certificates. Although they are based on different cryptographic techniques, their underlying processes of creation and use are similar. Particularly, pseudonymous certificates allow providing both authentication and privacy protection[4].

4.4. Creation of pseudonymous certificates

Pseudonymous certificates must be issued by a trusted authority. A Vehicular Public Key Infrastructure (VPKI) is often assumed for this purpose. Fig 3 shows its composition and its relationships with other entities that were introduced on the VANET model. VPKI is composed by a set of Trusted Third Parties (TTPs) in charge of managing pseudonymous certificates. It is assumed to be structured hierarchically. There is a single root Certificate Authority (CA) in each administrative domain (e.g. a country) and a delegated CA in each region within that domain. As vehicles from different regions (or even domains) can encounter themselves in a VANET, it is generally assumed that these CAs will be mutually recognized.

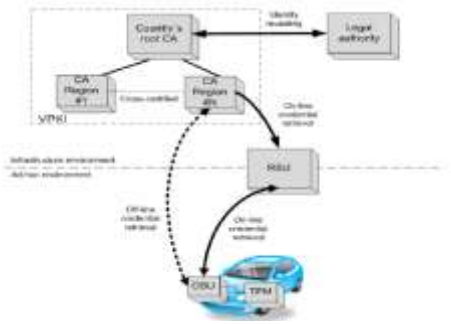

Figure 3: Alternatives to retrieve vehicular credentials

Taking the need of resolvable anonymity into account, there must be a relationship between the vehicle´s real identity and each of its pseudonyms. In fact, as reports are issued to people (and not to vehicles), there are two different steps to link the pseudonym with the vehicle owner´s real identity. The relationship between ELP and pseudonym is managed by the VPKI, whereas the link between ELP and the owner´s identity is only known by the legal authority. Once misbehavior is detected, the authority will contact VPKI in order to get the ELP related to a specific pseudonym. As this identity resolution removes the privacy protection, this process has to be performed only when necessary [4].

4.5. Use of pseudonymous certificates

To harden tracking, each credential should not be used for a long time. Thus, a change policy should be established. Nevertheless, the process of pseudonym change is far from trivial. Its effectiveness is directly related to how difficult would be for an attacker to link both pseudonyms (i.e. the former and the new pseudonyms). Mix contexts have been proposed to perform such changes these areas are unmonitored by any RSU and are preferably put in road intersections. All communications are stopped while being inside that area. Vehicles may change their pseudonyms before leaving it. In this way, when many vehicles enter on this area, their new pseudonym is difficult to guess when they left the mix context.

4.6. Information trust

Every vehicle has to check the reliability of the received messages. Apart from checking the used cryptographic values (if any), it has to evaluate if the contained information could be true.

4.7. Verification by correlation

In the bogus information attack, one or several legitimate members of the network send out false information to misguide other vehicles about traffic conditions. To thwart such misbehavior, data received from a given source should be verified by correlating them with those received from other sources. This is typically done by reputation-based systems. It is important to stress here that what matters is the rating of the correctness of the data rather than its source.

V. Simulators For Vanets

Deploying and testing VANETs involves high cost and intensive labor. Hence, simulation is a useful alternative prior to actual implementation. Simulations of VANET often involve large and heterogeneous scenarios.

We have classified existing VANET simulation software into three different categories see Fig 4.. They are (a) vehicular mobility generators, (b) network simulators, and (c) VANET simulators. Vehicular mobility generators are needed to increase the level of realism in VANET simulations. They generate realistic vehicular mobility traces to be used as an input for a network simulator. Network simulators perform detailed packet-level simulation of source, destinations, data traffic transmission, reception, background load, route, links, and channels. VANET simulators provide both traffic flow simulation and network simulation.

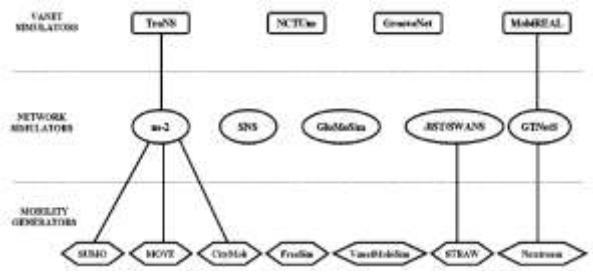

Figure 4: A taxonomy of VANET simulation software.

5.1. Mobility Generators

5.1.1.VanetMobiSim [5] is an extension of the CANU Mobility Simulation Environment (CanuMobiSim) which focuses on vehicular mobility, and features realistic automotive motion models at both macroscopic and microscopic levels. At the macroscopic level, VanetMobiSim can import maps from the US Census Bureau topologically integrated geographic encoding and referencing (TIGER) database, or randomly generate them using Voronoi tessellation. The TIGER/Line files constitute a digital database of geographic features, such as roads, railroads, rivers, lakes, and legal boundaries, covering the entire United States. VanetMobiSim adds support for multi-lane roads, separate directional flows, differentiated speed constraints and traffic signs at intersections. At the microscopic level, it supports mobility models such as Intelligent Driving Model with Intersection Management (IDM/IM), Intelligent Driving Model with Lane Changing (IDM/LC) and an overtaking model (MOBIL), this interacts with IDM/IM to manage lane changes and vehicle accelerations and decelerations, providing realistic car-to-car and car-to-infrastructure interactions. VanetMobiSim is based on JAVA and can generate movement traces in different formats, supporting different simulation or emulation tools for mobile networks including ns-2, GloMoSim , and QualNet .

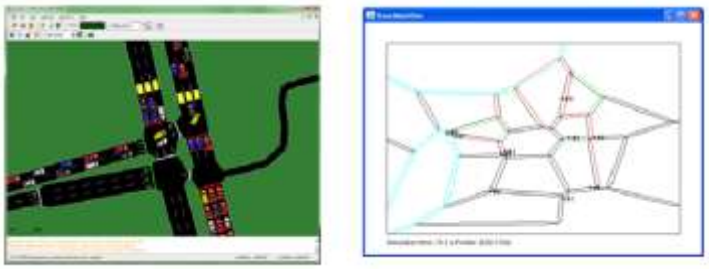

Figure 5 :GUI of (a) SUMO simulator (b) Vanetmobisim

5.1.2.SUMO (Simulation of Urban MObility) [6] is an open source, highly portable, microscopic road traffic simulation package designed to handle large road networks. Its main features include collision free vehicle movement, different vehicle types, single-vehicle routing, multi-lane streets with lane changing, junction-based right-of-way rules, hierarchy of junction types, an openGL graphical user interface (GUI), and dynamic routing. SUMO can manage large environments, i.e., 10 000 streets. Thus, by combining SUMO and openstreetmap.org , we can simulate traffic in different locations of the globe. However, since SUMO is a pure traffic generator, its generated traces cannot be directly used by the available network simulators, which is a serious shortcoming.

5.1.3. MOVE (MObility model generator for Vehicular networks) [7] rapidly generates realistic mobility models for VANET simulations. MOVE is built on top of SUMO. The output of MOVE is a mobility trace file that contains information of realistic vehicle movements which can be immediately used by popular network simulation tools such as ns-2 or GloMoSim. In addition, MOVE provides a GUI that allows the user to quickly generate realistic simulation scenarios without the hassle of writing simulation scripts as well as learning about the internal details of the simulator.

5.1.4. STRAW (STreet RAndom Waypoint) provides accurate simulation results by using a vehicular mobility model on real US cities, based on the operation of real vehicular traffic. STRAW's current

implementation is written for the JiST/SWANS discrete-event simulator, and its mobility traces cannot be directly used by other network simulators, such as ns-2. STRAW is part of the C3 (Car-to-Car Cooperation) project . A more realistic mobility model with the appropriate level of detail for vehicular networks is critical for accurate network simulation. The STRAW mobility model constrains node movement to streets defined by map data for real US cities and limits their mobility according to vehicular congestion and simplified traffic control mechanisms.

5.1.5.FreeSim is a fully customizable macroscopic and microscopic free-flow traffic simulator that allows for multiple freeway systems to be easily represented and loaded into the simulator as a graph data structure with edge weights determined by the current speeds. Traffic and graph algorithms can be created and executed for the entire network or for individual vehicles or nodes, and the traffic data used by the simulator can be user generated or be converted from real-time data gathered by a transportation organization. Vehicles in FreeSim can communicate with the system monitoring the traffic on the freeways, which makes FreeSim ideal for Intelligent Transportation System (ITS) simulation. FreeSim is licensed under the GNU General Public License, and the source code is available freely for download.

5.1.6.CityMob v.2 CityMob is a ns-2 compatible mobility model generator proposed for use in VANETs. Citymob implements three different mobility models: (a) Simple Model (SM), (b) Manhattan Model (MM), and (c) realistic Downtown Model (DM). In DM model, streets are arranged in a Manhattan style grid, with a uniform block size across the simulation area. All streets are two-way, with lanes in both directions. Car movements are constrained by these lanes. Vehicles will move with a random speed, within an user-defined range of values. DMmodel also simulates semaphores at random positions (not only at crossings), and with different delays. DM adds traffic density in a way similar to a real town, where traffic is not uniformly distributed. Hence, there will be zones with a higher vehicle density. These zones are usually in the downtown, and vehicles must move more slowly than those in the outskirts. CityMob DM also has the following capabilities: (a) multiple lanes in both directions for every street, (b) vehicle queues due to traffic jams, and (c) the possibility of having more than a downtown.

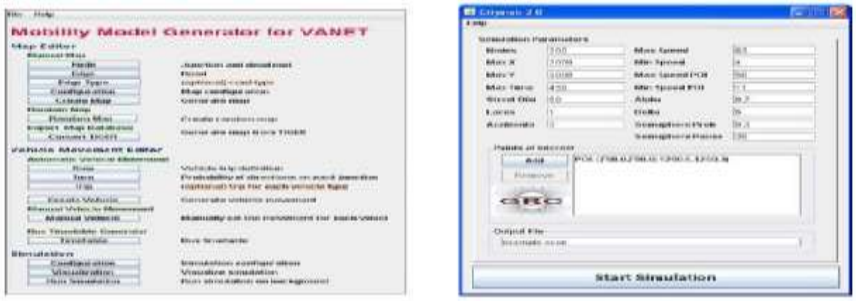

Figure 6: GUI of (a) MOVE (b) CityMob v.2.

Table 1:Comparison of mobility generators.

	VanetMobiSim	SUMO	MOVE	STRAW	FreeSim	CityMob
Software						
Portability	✓	✓	✓	✓	✓	✓
Freeware	✓	✓	✓	✓	✓	✓
Opensource	✓	✓	✓	✓	✓	✓
Console	×	✓	✓	—	×	✓
GUI	✓	✓	✓	✓	✓	✓
Available examples	✓	✓	—	—	✓	×
Continuous development	×	✓	×	×	—	✓
Ease of setup	Moderate	Moderate	Easy	Moderate	Easy	Easy
Ease of use	Moderate	Hard	Moderate	Moderate	Easy	Easy
Maps						
Real	✓	✓	✓	✓	✓	×
User defined	✓	✓	✓	—	×	×
Random	✓	✓	✓	×	×	✓
Manhattan	×	×	×	×	×	✓
Voronoi	✓	×	×	×	×	×
Mobility						
Random waypoint	✓	✓	✓	×	×	✓
STRAW	×	✓	✓	✓	×	×
Manhattan	×	✓	✓	×	×	✓
Downtown	×	×	×	×	×	✓
Traffic models						
Macroscopic	×	×	×	×	✓	×
Microscopic	✓	✓	✓	✓	✓	✓
Multilane roads	✓	✓	✓	✓	—	✓
Lane changing	✓	✓	✓	✓	—	✓
Separate directional flows	✓	✓	✓	✓	—	✓
Speed constraints	✓	✓	✓	✓	✓	✓
Traffic signs	✓	✓	✓	✓	—	✓
Intersections management	✓	✓	✓	—	—	×
Overtaking criteria	✓	—	—	—	—	×
Large road networks	—	✓	✓	✓	—	✓
Collision free movement	—	✓	✓	—	—	✓
Different vehicle types	×	✓	✓	—	×	✓
Hierarchy of junction types	×	✓	✓	—	×	×
Route calculation	✓	✓	✓	✓	✓	×
Traces						
ns-2 trace support	✓	×	✓	×	×	✓
GloMoSim support	✓	×	✓	×	×	×
QualNet support	✓	×	✓	×	×	×
SWANS support	×	×	×	✓	×	×
XML-based trace support	✓	×	×	×	×	×
Import different formats	✓	✓	✓	×	×	×

5.2. Network Simulators

5.2.1.NS-2 [8] is a discrete event simulator developed by the VINT project research group at the University ofCalifornia at Berkeley. The simulator was extended by the Monarch research group at Carnegie Mellon University to include: (a) node mobility, (b) a realistic physical layer with a radio propagation model, (c) radio network interfaces, and (d) the IEEE 802.11 *Medium Access Control* (MAC) protocol using the distributed coordination function (DCF).

5.2.2. GloMoSim [9] is a scalable simulation environment for wireless and wired network. It has been designed using the parallel discrete-event simulation capability provided by Parsec. GloMoSim has been built using a layered approach similar to the OSI seven layer protocol models. Standard APIs are used between the different simulation layers. This allows the rapid integration of models developed at different layers by different people. The widely used QualNet simulator is a commercial version of GloMoSim.

5.2.3. JiST/SWANS [10]. JiST is a high performance discrete event simulation engine that runs over a

Figure 7: GUI of NS2 simulator

standard Java virtual machine. It is a prototype of a new general purpose approach to building discrete event simulators, that unifies the traditional systems and language-based simulator designs. It outperforms existing highly optimized simulation engines both in time and memory consumption. Simulation code that runs on JiST need not be written in a domain-specific language invented specifically for writing simulations, nor must it be littered with special purpose system calls and 'call backs' to support runtime simulation. Instead, JiST converts an existing virtual machine into a simulation platform, by embedding simulation time semantics at the byte-code level. Thus, JiST simulations are written in Java, compiled using a regular Java compiler, and run over a standard, unmodified virtual machine. SWANS is a scalable wireless network simulator built on top of the JiST platform. It was created primarily because existing network simulation tools are not sufficient for current research needs.SWANS contains independent software components that can be composed to form complete a wireless network or sensor network. Its capabilities are similar to ns-2 and GloMoSim, but SWANS is able of simulating much larger networks. SWANS leverages the JiST design to achieve higher simulation throughput, lower memory requirements, and run standard Java network applications over simulated networks.

5.2.4. SNS (a Staged Network Simulator) [11]. Traditional wireless network simulators are limited in speed and scale because they perform many redundant computations both within a single simulation run, as well as across multiple invocations of the simulator. The staged simulation technique proposes to eliminate redundant computations through function caching and reuse. The central idea behind staging is to cache the results of expensive operations and reuse them whenever possible. SNS is a staged simulator based on ns-2. On a commonly used ad hoc network simulation setup with 1500 nodes, SNS executes approximately 50 times faster than regular ns-2 and 30% of this improvement is due to staging, and the rest to engineering. This level of performance enables SNS to simulate large networks. However, the current implementation is based on ns-2 version 2.1b9a, and it is not specifically designed to simulate VANET scenarios.

Table 2: Comparison of network simulators

	ns-2	GloMoSim	JIST/SWANS	SNS
Software				
Portability	√	√	√	√
Freeware	√	√	√	√
Opensource	√	√	√	√
Available examples	√	√	√	√
Continuous development	√	x	√	x
Large networks	x	√	√	√
Console	√	√	√	√
GUI	√	√	√	√
Scalability	Poor	High	High	High
Ease of setup	Easy	Moderate	Hard	Easy
Ease of use	Hard	Hard	Hard	Hard
VANET				
802.11p	Only for ns-2.33	x	x	x
Obstacles	x	x	x	x
Vehicular traffic flow model	x	x	x	x

5.3. VANET Simulators

5.3.1. TraNS (Traffic and Network Simulation Environment) [12] is a simulation environment that integrates both a mobility generator and a network simulator and it provides a tool to build realistic VANET simulations. TraNS provides a feedback between the vehicle behavior and the mobility model. For example, when a vehicle broadcasts information reporting an accident, some of the neighboring vehicles may slow down. TraNS is an open open-source project providing an application-centric evaluation framework for VANETs. TraNS is written in Java and C++ and works under Linux and Windows (trace-generation mode). The current implementation of TraNS uses the SUMO traffic simulator and the ns-2 network simulator. It is being developed at EPFL, Switzerland.

5.3.2. GrooveNet [13] is a hybrid simulator which enables communication between simulated vehicles and real vehicles. By modeling Inter- Vehicular Communication within real street map based topography, it eases protocol design and in-vehicle deployment. GrooveNet's modular architecture incorporates mobility; trip and message broadcast models over a variety of link and physical layer communication models. GrooveNet supports simulations of thousands of vehicles in any US city as well as the addition of new models for networking, security, applications, and vehicular interactions. It provides multiple network interfaces, and allows GPS and event-triggered (from the vehicles' onboard computer) simulations.

5.3.3. NCTUns (National Chiao Tung University Network Simulator) [14] is a high-fidelity and extensible network simulator and emulator capable of simulating various protocols used in both wired and wireless IP networks. Its core technology is based on a novel kernel re-entering methodology. Due to this novel methodology, NCTUns provides many unique advantages that cannot be easily achieved by traditional network simulators such as ns-2 and OPNET. The NCTUns network simulator and emulator has many useful features. It can be easily used as an emulator since it supports seamless integration of emulation and simulation. It uses Linux TCP/IP protocol stack to generate high-fidelity simulation results. It can run any real-life UNIX application program on a simulated node without any modifications. Supported networks include Ethernet-based fixed Internet, IEEE 802.11b wireless LANs, IEEE 802.11e QoS wireless LANs, IEEE 802.16d WiMAX wireless networks, DVBRCS satellite networks, wireless vehicular networks for Intelligent Transportation Systems (including V2V and V2I), multi-interface mobile nodes for heterogeneous wireless networks, IEEE 802.16e mobile WiMAX networks, IEEE 802.11p/1609WAVEwireless vehicular networks, etc.

Figure 8: GUI of (a) GrooveNet, (b) NCTUns

Figure 9: GUI of (a) TraNS. (b) MobiREAL

5.3.4. MobiREAL [15] provides a new methodology to model and simulate realistic mobility of nodes and evaluate MANET applications. It is a network simulator that can simulate realistic mobility of humans and vehicles, and allow the changing of their behavior depending on a given application context. MobiREAL can easily describe mobility of nodes using C++. It adopts a probabilistic rule based model to describe the behavior of mobile nodes, which is often used in cognitive modeling of human behavior. The proposed model allows one to describe how mobile nodes can change their destinations, routes and speeds/directions based on their positions, surroundings (obstacles and neighboring nodes), and information obtained from applications.

Table 3: Comparison of VANET simulators

	TraNS	GrooveNet	NCTUns	MobiReal
Mobility generator Network simulator	SUMO ns-2	GrooveNet	NCTUns	MobiReal based on GTNetS
Mobility models	Random and manual routes	Random waypoint, explicit origin-destination, distributed origin-dest	Random and manual routes	probabilistic rule-based
Simulation type	Microscopic, space-continuous and time-discrete			
Lane models	Multi-lane streets with lane changing			
Speed models	Street speed	Uniform, street speed, Markov model, load-based	Random	Street speed
Traffic flow model	Car following SK and traffic assignment using the DUA-approach	Car following	Car following	Car following
Road topology	Any	Any	User defined	Any
Traffic lights	Manually defined	Manually defined	Automatically generated on intersections	Manually defined
Intersection model	Junction-based right-of-way rules Hierarchy of junction types	Managed by traffic lights	Managed by four traffic lights	right-of-way rules and managed by traffic lights
Trip model	Random, manually defined	Dijkstra, sightseeing	Manually defined	Manually defined
VANET protocols and facilities	802.11p two ready-to-use VANET applications: road danger warning (safety) and dynamic reroute (traffic efficiency) tested up to 3000 vehicles	Supports V2V and V2I communications multiple message types, which are broadcast periodically to inform neighbors of a vehicle's current position, and vehicle emergency with priorities	802.11p, supports multiple interfaces at the same time car agents control the driving behavior moving on a road.	Initially it was especially designed for MANETs instead of VANETs.
VANET built-in application support	Road danger warning and dynamic reroute	Vehicle warning and adaptive rebroadcast	None	None
Ease of setup	Moderate	Moderate	Hard	Easy
Ease of use	Moderate	Hard	Hard	Hard
Comments	Integrates both traffic and network simulators. Information exchanged in communication protocols can influence the vehicle behavior in the mobility model.	Able to support hybrid simulations (i.e., communication between simulated vehicles and real vehicles on the road)	Supports seamless integration of emulation and simulation, but it needs Fedora nine Operating System to be installed	Simulates realistic mobility of humans and cars, and their behavior can be changed depending on the given application context

VI. Conclusion

In this section we have reviewed existing routing protocols, security issues and simulation tools. Routing is an important component in vehicle-to-vehicle (V2V) and infrastructure-to-vehicle (I2V) communication. This paper discusses various routing protocols of VANET. Vehicular Ad Hoc Networks is an emerging and promising technology, this technology is a fertile region for attackers, who will try to challenge the network with their malicious attacks. The increasing popularity and attention in VANETs has prompted researchers to develop accurate and realistic simulation tools. In this paper, we make a survey of several publicly available mobility generators, network simulators, and VANET simulators.

REFERENCES

[1]. Yue Liu, Jun Bi, Ju Yang; "Research on Vehicular Ad Hoc Networks"; Chinese Control and Decision
[2]. Conference (CCDC), 2009. Page(s): 4430 – 4435.
[3]. M Raya, J Pierre Hubaux," The security of VANETs", Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks, 2005.
[4]. B. Parno and A. Perrig, "Challenges in Securing Vehicular Networks", Proc. of HotNets-IV, 2005.
[5]. José María de Fuentes, Ana Isabel González-Tablas, Arturo Ribagorda "Overview of security issues in Vehicular Ad-hoc Networks", Handbook of Research on Mobility and Computing
[6]. Haerri J, Fiore M, Fethi F, Bonnet C. VanetMobiSim: generating realistic mobility patterns for VANETs. Institut Eurécom and Politecnico Di Torino, 2006.
[7]. Krajzewicz D, Rossel C. Simulation of Urban Mobility (SUMO). German Aerospace Centre, 2007. Available at: http://sumo.sourceforge.net/index.shtml MOVE (MObility model generator for VEhicular networks): Rapid Generation of Realistic Simulation for VANET, 2007.
[8]. Fall K, Varadhan K. ns notes and documents. The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC, February 2000. Available at: http://www.isi.edu/nsnam/ns /nsdocumentation. html
[9]. Martin J. GloMoSim. Global mobile information systems simulation library. UCLA Parallel Computing Laboratory, 2001. Available at: http://pcl.cs.ucla.edu/projects/glomosim/
[10]. JiST/SWANS: Java in Simulation Time/Scalable Wireless Ad hoc Network Simulator, 2004. Available at: http://jist.ece./cornell.edu/
[11]. Walsh K, Sirer EG. A staged network simulator (SNS). Computer Science Department, Cornell University, 2003. Available at: http://www.cs.cornell.edu/people/egs/sns/
[12]. Piorkowski M, Raya M, Lugo AL, Papadimitratos P, Grossglauser M, Hubaux J-P. TraNS (Traffic and Network Simulation Environment). Ecole Polytechnique Fédérale de Lausanne, EPFL, Switzerland, 2007. Available at: http://trans.epfl.ch/
[13]. Mangharam R, Weller D, Rajkumar R, Mudalige P, Bai F. GrooveNet: A Hybrid Simulator for Vehicle-toVehicleNetworks.CarnegieMellonUniversity,2006.Availableat:http://www.seas.upenn.edu/rahulm/Research/GrooveNet/
[14]. NCTUns 5.0, 2008. Available at: http://nsl10.csie.nctu.edu.tw/
[15]. MobiREAL, 2008. Available at: http://www.mobireal.net/

Bibilography of the Author

Mushtak Y. Gadkari , Lecturer in information technology ,in Rajendra mane college of engineering and technology,Ambav having 5 years of experience in teaching. Persuing ME in Electronics & Telecommunication. Research area is VANET and Wireless network

Nitin B. Sambre ,Assistant professor in KIT, Kolhapur,having 12 years of teaching experience . his area of interest is networking and image processing