Shaima M. Shakeel[1], Osama M. H. Rehman[1], Mohamed Ould-Khaoua[2], Hadj Bourdoucen[1]

Open Source Software Support for Field Experiments of Vehicular Ad Hoc Networks

Abstract—**The ever increasing need of technology-users to have greater levels of stability, control and security over the hardware has paved a long way in the field of free and open-source software (FOSS). The FOSS-based powerful tools have a plethora of uses in the field of communication. Vehicular ad hoc networks (VANETs) is a part of the intelligent transportation system(ITS), which is a rapidly growing area of research since the past decade.VANETs aim to provide better traffic management, such as the reduction ofroad accidents and traffic congestion. This paper sheds some light on the usage of FOSS tools to conduct field tests in VANETs. Various open source tools such as Linux operating system, C programming language, AWK scripting and Wireshark network packet analyzer have been deployed to study and evaluate VANETs. This paper also presents the benefits and drawbacks faced while using the FOSS tools to carry out the field experiments on VANETs.**

Index Terms— **C++, Free and open source software (FOSS), Linux, VANETs, Wireshark.**

XLIII. INTRODUCTION

Vehicular Ad Hoc NETworks (VANETs) is considered as one of the state-of-the-art field of research [1]. Since the mankind has begun to witness the development of roads and means of transportation to ease and comfort the traveling experience, it has been inevitable to hear of negative effects of it as well. In other words, these developments made in transportations, in order to ease our lives, goes hand in hand with being a great source of physical, financial and psychological risks.

Recently, the field of vehicular communication and networking had gained a momentum for development [2]. With the recent technological advances in various fields, a new trend of communication referred to as the ad-hoc network has come into existence.This is a special type of decentralized wireless networks, where each communicating device in the network (referred to as a node) has the property of arranging itself and acting as an end-node and as a router in a very dynamic fashion, as and when required. One of the types of ad-hoc networks is the Mobile Ad-Hoc NETworks (MANETs), where nodes are mobile and free to move independently and hence the links between these nodes also alters frequently [3,4].

[1]Communication and Information Research center,Department of Electrical & Computer Engineering, Sultan Qaboos University, P.O. 17, Al-Khodh, Muscat-123, Sultanate of Oman (phone: 968-24143696; fax: 968-24144336; e-mail: m100170@student.squ.edu.om,p091608@student.squ.edu.om, hadj@squ.edu.om)

[2]Department of Electrical & Computer Engineering, College of Engineering, Sultan Qaboos University, P.O. 33, Al-Khodh, Muscat-123, Sultanate of Oman (phone: 968-24142537; fax: 968-24413454; e-mail: mok@squ.edu.om)

A subcategory of MANETs which has recently been a hot topic for research is the area where the MANETs span the vehicular movement, such as road vehicles, automobiles, trains etc. This important subcategory is classified in general as VANETs.

VANETs are used to facilitate inter-vehicle communication with the intention of improving the road safety, efficiency and comfort of the road journey in day-to-day life. This type of communication can work without the need for expensive infrastructure. VANETs rely on direct communication between vehicles to satisfy the communication needs of a large class of applications (e.g., collision avoidance, passing assistance and platooning). VANETs systems can be supplemented or, in some situations, replaced by roadside infrastructure, allowing for Internet access and several other applications [5].

In VANETs, each vehicle is considered as a node which will act as a router to transmit or receive information from the other nodes on the road. An on-board unit (OBU) within the car will help the car to receive warnings for the driver and act accordingly to prevent accidents. Each car can have a range of communication (100m to 1000m) depending on the power or transmission, to send or receive information [6]. In such a situation, the other cars that join in to form a new network and act as intermediate nodes to transmit or receive the signal, and hence, a mobile network is created.In order to study VANETs under different scenarios, two methods have been most widely used:

1) Vehicular ad hoc networks *Simulation*: Various VANET simulation softwares are free and open source such as the traNS, NCTUns, GrooveNet, MobiREAL are used. In addition, the network simulators which are most commonly used are ns2, ns-3, SNS, GloMoSim and GTNetS[7].
2) *Field testing/ prototyping*: where the actual deployment of the technologies are put to test in real life scenarios on the vehicles on road use the required equipment for communicating between on-road vehicles

Various research projects have given solutions for fulfilling the technological requirements in VANETs while using simulations software. However, current focus is shifting from simulated-based research to practical implementation in the form of on-field operational tests [8,9]. The authors in [10,11] have carried out various field tests to evaluate the vehicle-to-vehicle and vehicle-to-infrastructure applications in the real world scenario to prove that VANETs can indeed enhance road safety, traffic efficiency and travel comfort. Moreover, a number of researchers have designed an enhanced emergency warning system [12] and tested it in order to evaluate its performance.

This paper aims to demonstrate the effectiveness of expending the FOSS tools in performing field testing in VANETS and discusses some of the short comings faced in the process of deploying these open source tools.

The rest of the paper is organized as follows. Section II discusses the arrangement of the testing environment and provides a thorough description of the FOSS tools used in the implementation of field experiments. Section III briefly discusses and comments on the results obtained from testing in the static nodes scenario. Section IV highlights some of the shortcomings faced by the deployment of the FOSSC tools in these experiments. Finally, Section V mentions the challenges faced during this study and the future works to be conducted in this study.

XLIV. FOSS TOOLS IN VANETS TEST ENVIRONMENT

A. FOSS Tools

1) Linux operating System

The popularity of this operating system can be justified by the fact that it is capable of supporting a host of hardware platforms including audio & video devices, network cards, GPS, laptops and notebooks, motherboards, cameras and antennas to mention a few [13]. It also provides an efficient platform to run open source network simulation softwares such as NS2 [14]. Another significant feature of Linux based operating systems is the ability to create and support efficient links in wireless Ad Hoc networks with ease. And hence, all these features make the operating system a suitable FOSS tool for performing tests in VANETs.

The field tests conducted in this study has been implemented using the latest versions Ubuntu, namely: 12.04 and 14.04. Ubuntu is free and open source software which works on a Linux based kernel.

The equipment and the devices used for the tests are tabulated in Table 1.

Table 1: List of Devices used to Field tests

Hardware Device	Device Brand	Device Model
2.4GHz 8dBi Indoor Desktop Omni-directional Antenna	TP-Link	TL-ANT2408C
USB Wireless WiFi Network Adapter	Alfa	AWUS036H
Mouse GPS	Globalsat	BU-353S4
Data Processing Units (Laptop)	Toshiba	Satellite C660-24E

2) C Language & AWK scripting language

The C language is the base of all the coding involved in the kernel and auxiliary softwares developed in Linux based operating system. It was essentially created for the purpose of designing system based and embedded software programming. It is the most commonly used languages for designing and building open source software tools [15].

AWK is yet another built in feature of Linux operating systems, which is a data extraction tool written for manipulating streams of textual data [16]. It is a very flexible and efficient tool when it comes to processing and extracting specific set of data from a large chunk of pipelined data or a large data file.

For the field tests on VANETS, socket programming was used for inter process communication protocol between sender and receiver nodes in the network. There are two types of sockets that are commonly used: TCP (stream) sockets and UDP (Datagram) sockets. Another type of socket is the RAW socket, which allows direct access to lower-layer protocols, such as Internet Protocol (IPv4 or IPv6) and Internet Control Message Protocol (ICMP or ICMP6). [17].

C language has been used for creating the socket programming in the field experimental tests conducted in this work. It has been compiled using the GCC C compiler, which is free and open source software.

3) Wireshark packet Tracer

Wireshark is a graphical user interface based FOSS that was developed for the analysis of network packets [18]. Besides being an open-source tool, Wireshark is a cross-platform software that runs on many platforms including Microsoft and GNU/Linux. Wireshark has the capability to understand the structure of different networking protocols, where it can analyze and display the fields of the protocols. Based on the above discussed Wireshark's capabilities, the tool has the potential for analyzing the received periodic Hello packets exchanged between nodes in a vehicular network. Authors in [19] used the Wireshark tool in their on-field experiments on vehicular obstructions in VANETs, in order to analyze the PDR and received signal strength indicator (RSSI).

B. Test Environment

1) Hardware:

The experiments have been performed using two sedan vehicles, each of which acts as a node in the vehicular network. Each vehicles is equipped with a laptop (as a data processing unit), a WiFi network adapter, an Omni directional antenna and a mouse GPS.

The laptops were running Ubuntu 14.04 operating system and were connected with the antenna to extend and improve the wireless range and performance. Since the antenna could not be directly connected to the laptop, a wireless USB network adapter was used as an interface between the two.

In order to find the position of the vehicles, the mouse GPS was used to read the latitude and longitude coordinates of the same.

2) Software programs:

An effective, flexible and reliable programming code plays a major role in yielding effective results during field experiments. The programming needed for performing experimental field tests on VANETs was accomplished by developing an efficient socket program. A socket is adopted to act as an Application Programming Interface (API) between two different and simultaneously running processes across an ad hoc network. In the experiments done, the two processes running include the process running at the sender end, and the other process running at the receiver end.

The type of socket used is UDP socket, which is connectionless datagram socket. Each of the packets, which is a “Hello” packet is broadcasted to every other node in the ad hoc network and receives packets from every node in the network. Since only two nodes have been used, the packets are uninterruptedly being broadcasted and received to and from one node to the other. In addition to the socket program written in C language, number of file handling functions were also implemented to save the number of packets being sent and received at each node, in order to calculate the Packet Deliver Ratio of one node with respect to the other. Another important function deployed in the code was the calculation of the distance between the two nodes by using a mathematical haversine formula (1-3) based on the GPS coordinates of the receiver.

$$a = \sin^2(\Delta\varphi/2) + \cos \varphi 1 \cdot \cos \varphi 2 \cdot \sin^2(\Delta\lambda/2) \quad (1)$$

$$c = 2 \cdot \mathrm{atan2}(\sqrt{a}, \sqrt{(1-a)}) \quad (2)$$

$$d = R \cdot c \quad (3)$$

where φ is latitude, λ is longitude, R is earth's radius (mean radius = 6,371km).

With the reception of each Hello packet, the calculated distance between the two nodes is saved in a file along with the corresponding coordinates and MAC addresses of the sending node. This is further on used to test the accuracy of the distance measured by the GPS coordinates formula with respect to the actual distance, measured by a laser range finder (LRF). The difference in distance between the two discussed techniques is shown in section III.

The GPS receiver connected at each node receives information in the NMEA format, which is constantly saved into a file. Then the AWK scripting language is used to extract the location, and UTS time of the receiver node and the sender node.

The data being sent in each of the packets comprises of the location coordinates in latitude and longitude, the MAC address of the sender/receiver and the corresponding UTC time of the GPS clock, and the packet headers. The total size of each packet flowing through the network was 512 bytes. A total of 5000 packets were sent from both the nodes for each experiment, and the number of receiving packets is noted.

3) Field setup:

The nodes have been placed in a static (fixed) mobility mode scenario, each of them equipped with the devices mentioned in Table 1. The GPS was configured at each node to give the required NMEA format at a baud rate of 4800. For each of the experiments, all other parameters were kept fixed, and the distance between the two vehicles was being varied from the range of 10m to 450m, where distances were measured using aTruPulse 360BLRF. While measuring the distance, it was ensured that no obstacle, such as a vehicle or a building is present between the two nodes. Fig. 1 depicts the arrangement of the equipment listed in table 1. Each of the vehicles is installed with equipment in a similar fashion as shown in the figure.

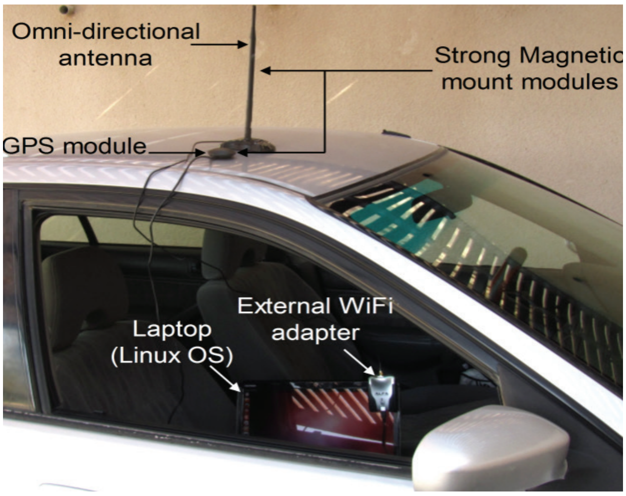

Fig. 1.Arrangement of equipment at each node (vehicle)

XLV. RESULTS

Two kinds of on-field experiments are performed in this work. The first experiment presents the accuracy of the measured inter-nodes distance,in a fixed nodes scenario, through the assistance of exchanged GPS coordinates. Conversely, the second experiment depicts the PDR fall as a function of distance from source.

In the first experiment, it was observed that the GPS-based measured inter-node distance did not produce a very significantly large error as compared to the LRF-based measured distance, as shown in Fig. 1.The experiments included inter-node distances varying from 10m to 450m.Fig. 1 presents both the magnitude of distance difference and the difference in percentage from the LRF-based measured distance.

The absolute positioning achieved by the GPS receivers can vary with time. This phenomenon is usually noticed due to a number of time varying effects. The effects includes the absence or availability of satellites and weather conditions controlling clarity of the sky, hence the line-of-sight (LOS) communication with the satellites. As a result of varying absolute positioning, the inter-node distance may vary for consecutively measured distances. However, as results depict in Fig. 1, the margin of error for all six experiemnts did not exceed more than 6%.

The second experiment shows the achieved PDR as a function of GPS-based distance. The distance is measured by the Haversine formula with assistance of the exchanged GPS coordinates. The coordinatesare recorded in each experiment with the considered varying distances. As shown in Fig. 2, the PDR followed a degrading trend as the distance increased. The observed phenomenon is as a result of increasing packet drops as distance increases from the broadcasting source.

The broadcasted messages have the probability of either being received or lost, which can be due to packets collisions, path loss and fading channels effects. The effects of packets collisions can be observed at any distance from the source. However, the effects of path loss and fading channel effects is more visible at larger distances from the source, as seen in Fig. 2. The PDR at a distance of 450m is approximately 28% in contrast to the 98% achieved at a distance of 10m. As observed in Fig. 2, a rapid fall in PDR is observed after distance of 250m from source indicating that higher transmission power are required for achieving better PDR performance at distances over 250m.

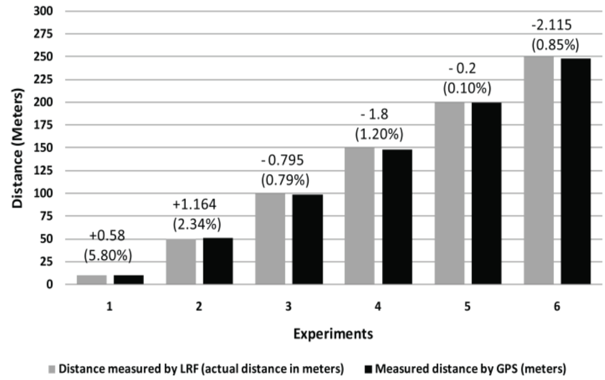

Fig. 2. Error margins in inter-node distance for fixed nodes in VANETs

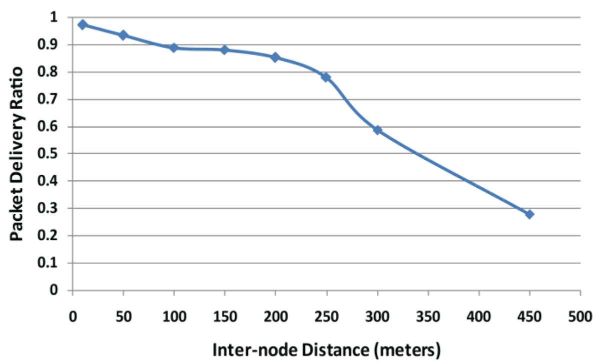

Fig. 3PDR as function of distance in VANETs on-road experiments

XLVI. Challenges & Future Work

Ashort coming of using FOSS tools was the GPS configuration with the Linux based system. Each time, the GPS dongle was mounted; it required a series of configuration steps to be followed before the GPS could receiver could start receiving signals from the GPS satellites and output an appropriate NMEA format output.

The future work would include calculating the average transmission delay of sending the packets over the network. And the node density will be increased to four nodes and further intensive tests will be performed in a dynamic mobility scenario with varying parameters including speed, packet size, and frequency of transmission.

XLVII. Conclusions

This paper has described the useof FOSS tools in implementing field tests for Vehicular Ad Hoc networks. These tools have proven to be a promising approach towards the implementation and provision of a considerably flexible and reliable method to validate and compare the results obtained through simulations of VANETs and deployment in actual road scenarios. Using FOSS tools in the VANET field experiments provides the user with flexibility to amend the software programs according to their requirements. Thus, the future of field tests in VANETs can be taken further ahead towards designing and testing efficient routing protocols and implementing several other road safety applications.

Acknowledgment

This paper is based upon research work supported by The Research Council (TRC) Oman, under Grant No: ORG/ICT/12/002, (www.trc.gov.om).

This research has used resources of the Free & Open Source Software (FOSS) Lab, under the Communication & Information Research Center (CIRC) at Sultan Qaboos University (SQU) - Oman, (www.squ.edu.om/circ/).

The authors would like to acknowledge the Information Technology Authority (ITA) - Oman, (www.ita.gov.om), for the support in establishing the FOSS lab at SQU.

References

[41] Jakubiak, Jakub, and YevgeniKoucheryavy. "State of the art and research challenges for VANETs." *Consumer Communications and Networking Conference, 2008*. CCNC 2008. 5th IEEE. IEEE, 2008.

[42] Toor, Yasser, Paul Muhlethaler, and AnisLaouiti. "Vehicle ad hoc networks: Applications and related technical issues." *Communications Surveys & Tutorials,* IEEE 10.3 (2008): 74-88.

[43] Qian, Yi, and Nader Moayeri. "Design of secure and application-oriented VANETs." *Vehicular Technology Conference*, 2008. VTC Spring 2008. IEEE. IEEE, 2008.

[44] Bansal, Meenakshi, Rachna Rajput, and Gaurav Gupta. "Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations." *The Internet Society* (1999).

[45] Mihail L. Sichitiu et al. "Inter-Vehicle Communication systems: A Survey", *IEEE Communications surveys*, 2nd Quarter 2008, Volume 10, No.2. pp-88-105.

[46] Al-Sultan, S., Al-Doori, M. M., Al-Bayatti, A. H., &Zedan, H. (2014). A comprehensive survey on vehicular Ad Hoc network. *Journal of network and computer applications*, 37, 380-392.

[47] Fan Li and Yu Wang, "Routing in Vehicular Ad Hoc Networks: A Survey", *IEEE Vehicular Technology magazine*, JUNE 2007, University of North Carolina at Charlotte.

[48] Christian Wei, "V2X communication in Europe – From research projects towards standardization and field testing of vehicle communication technology", *Computer Networks*, Volume 55, Issue 14, 6 October 2011, Pages 3103–3119.

[49] Stubing, H. ;Bechler, M. ; Heussner, D. ; May, T., "simTD: a car-to-X system architecture for field operational tests [Topics in Automotive Networking]", *Communications Magazine, IEEE* (Volume:48 , Issue: 5), pp. 148 – 154, May 2010.

[50] Massimo Bertozzi, Alberto Broggi, Alessandra Fascioli, "Vision-based intelligent vehicles: State of the art and perspectives*", Robotics and Autonomous Systems* 32 (2000) 1–16.

[51] R. Stahlmann, A. Festag, A. Tomatis, I. Radusch, F. Fischer, "starting European field tests for car-2-x Communication: the drive c2x framework", *18th ITS World Congress and Exhibition 2011.*

[52] Buchenscheit, A., Schaub, F. ; Kargl, F. ; Weber, M., "A VANET-based emergency vehicle warning system", *Vehicular Networking Conference (VNC)*, 2009 IEEE, 28-30 Oct. 2009,pp. 1-8.

[53] Ubuntu Manpage: deb - Debian binary package format". Manpages.ubuntu.com. Retrieved 2013-07-05.

[54] Osama M. Hussain, HadjBourdoucen, and Mohamed Ould-Khaoua. "Impact of ns-2 as FOSS Simulation tool for research in vehicular ad-hoc networks." *Free & open source software conference (FOSSC-13).* Muscat, Oman. 2013.

[55] Love, R., Are, S. H. W., Linus, A. C., Kernels, L. V. C. U., & Begin, B. W. (2004). Linux Kernel Development Second Edition.

[56] Aho, Alfred V., Brian W. Kernighan, and Peter J. Weinberger. "Awk—a pattern scanning and processing language." Software: Practice and Experience 9.4 (1979): 267-279.

[57] http://www.csd.uoc.gr/~hy556/material/tutorials/cs556-3rd-tutorial.pdf

[58] G. Combs *et al.*, "Wireshark-network protocol analyzer," [Online]. Available: http://www.wireshark.org

[59] Meireles, R.; Boban, M.; Steenkiste, P.; Tonguz, O.; Barros, J., "Experimental study on the impact of vehicular obstructions in VANETs," *Vehicular Networking Conference (VNC), 2010 IEEE* , vol., no., pp.338,345, 13-15 Dec. 2010.