6-2017

A Lightweight Message Authentication Framework in the Intelligent Vehicles System

Mostafa El-Said
Grand Valley State University, elsaidm@gvsu.edu

Alexander Arendsen
University of Central Florida

Samah Mansour
Grand Valley State University, mansours@gvsu.edu

The International Journal of Engineering and Science (IJES)
|| Volume || 6 || Issue || 6 || Pages || PP 33-39 || 2017 ||
ISSN (e): 2319 – 1813 ISSN (p): 2319 – 1805

A Lightweight Message Authentication Framework in the Intelligent Vehicles System

[1]Mostafa El-Said, [2]Alexander Arendsen, [3]Samah Mansour
[1]*School of Computing and Information Systems Grand Valley State University Allendale, MI 49401-9403*
[2]*University of Central Florida Computer Science Dept 100 Weldon Boulevard, Sanford, FL 32773*
[3]*School of Computing and Information Systems Grand Valley State University Allendale, MI 49401-9403*

----------ABSTRACT----------
Intelligent Vehicles System (IVS) supports a wide variety of Advanced Driver Assistance System (ADAS) services such as vehicle visibility detection. In implementing this service, the message authentication is a vital design parameter that protects victim vehicles from being tricked into accepting false messages as legitimate ones and make a false decision based on the incoming message. However, implementing message authentication service is too expensive especially if vehicles, initially, don't trust each others or there is no certificate of authority in place.
In this research, we investigate the use of the Basic Safety Message (BSM) behavior over time as a metric to allow a receiving vehicle to anticipate at what distance it will continue to receive BSMs from within-range vehicles. Therefore, the victim vehicle would reject the BSM messages that fall outside its acceptance window.
Simulation experiments are setup to study the realistic behavior of the BSM messages in different environment characteristics including changing the vehicle size, number of road lanes and vehicle speed. Research findings suggested that the lightweight message authentication can assist vehicles in estimating the duration for a trusted relationship among those that are located within range of each others.
Keywords *- ITS, DSRC, message authentication, Visibility.*

Date of Submission: 14 January 2017 Date of Accepted: 09 June 2017

I. INTRODUCTION

In a connected vehicle system, Basic Safety Message (BSM) is used to convey road safety messages on US high ways [1]. Connected vehicles are communicating using either Vehicle-to-Vehicle (V2V) or Vehicle-to-Infrastructure communication architecture. V2V communications aims to provide each vehicle with 360-degree situational awareness of nearby vehicles, and by complementing onboard sensors in detecting possible crash scenarios [3]. Moreover, in V2V or V2I architecture, vehicles use Dedicated Short Range Communications (DSRC) transmissions to share Advanced Driver Assistance System (ADAS) safety information services such as road conditions or vehicle's location, weight, size or its heading direction, probability of vehicle collision with neighboring vehicles. Authors in [1], indicated that a recent report issued by the National Highway Traffic Safety Administration United States (NHTSA) shows that implementing just two V2V safety applications (Intersection Movement Assist (IMA) and Left Turn Assist (LTA)) will potentially prevent 25,000 to 592,000 crashes, save 49 to 1,083 lives, avoid 11,000 to 270,000 maximum abbreviated injuries, and minimize 31,000 to 728,000 property damage. Research efforts to better use the BSM message in the dissemination process of safety messages presented in the work presented by authors in [1]
Authors in [1], developed a Tractor-Trailer Basic Safety Message (TT-BSM) extensions to correctly represent the location of the articulated vehicles in V2X communications. The authors' approach aims to reduce the potential for false safety warnings messages in the DSRC-based systems. Authors indicated that the additional data sent in the BSM will have a minor effect on the over-the-air data traffic or the channel utilization efficiency because the number tractor-trailer vehicles is fairly a small fraction of the overall vehicles on the road. However, attacking and altering the safety messages sent in the V2V environment becomes possible due to the fact that vehicles have various penetration inputs. Consequently, this will lead to defeat the purpose of implementing a V2V communication to promote awareness and safety on our roads.
Vehicle attack surface represents all the possible ways to attack a victim vehicle. The target could be the vehicle's On Board Unit (OBU) or the entire vehicle system. The vehicle has multiple entry points to synchronize the attack that is represented by the system's possible weaknesses. In a V2V or V2I network architecture, an attacker will consider all the possible ways to inject malicious or fake data into the vehicle system via any of its sensors such as RADAR, LIDAR, ultrasound or the DSRC sensor such as shown in Fig. 1.

Furthermore, the attacker constructs a dynamic threat model to study and track the victim vehicle to learn about how these sensors communicate and at what time they are active. Then, choosing the location, entry point and the right time to inject the malicious data becomes a critical factor to execute the hacking activity [2].

Figure 1. V2V Attack Scenario

Efforts made research and industry communities to protect the BSM messages over the air. In [3], authors introduced an On-board Unit (OBU) VBSS solution, which generates a short-term certificate and signs Basic Safety Messages (BSM) to preserve privacy and enhance security. In this work, authors developed their solution based on the Public Key Infrastructure (PKI) where vehicles create groups. Each group will have its public and private keys, and a vehicle will sign a message using its group's private key. Moreover, authors encouraged research community to continue conducting further research to validate the proposed vehicle-based approach in V2V credential management.

In this research, we are investigating another Vehicle-Based Security System (VBSS) for a V2V system to provide security and privacy via developing a lightweight message authentication framework solution. The proposed framework allows a victim vehicle to discriminate against spoofed BSM messages. When building this framework, several design goals were considered and special efforts were made to:

- Provide on a vehicle infrastructure-less solution.
- Rely on a Certificate Authority-less solution. Therefore, no need to the use of certificate authority body hosted on a fixed Radio Service Unit (RSU).
- Predict the authentication session duration for how long the two vehicles will build their trust relationship via message authentication.

The proposed solution's framework consists of two phases such as described below:

Phase-I:

- In this phase, the authentication session lifetime is estimated by having two vehicles predicting a Visibility Time Window (VTW) factor based on the timing and the sequence of the BSM messages and the vehicle's speed. So, essentially the VTW window defines the duration for the authentication session. Simulation experiments will be conducted to determine the VTW in different simulation environments. Messages arrived outside the VTW may be ignored because they may redundant or malicious.

Phase-II: Monitoring the consistency of a

- In this phase, the sending vehicle will calculate a Message Authentication Code (MAC) and the receiving vehicle will verify this code over the duration of the VTW window such as follows:

a. At the sending vehicle, the MAC value is calculated based on hashing *the vehicle VIN number, location and size*. Then the sending vehicle sends out its BSM message along with the calculated MAC value.

b. At the receiving vehicle, the OBU verifies the BSM message integrity using the incoming MAC value as well as calculating a hash value such as follows:

i. Use the camera sensor to capture a picture for the sending vehicle,
ii. Use the picture estimation algorithm in [4] to estimate the vehicle's size,
iii. Hash the sending vehicle's estimated size from step (ii), VIN number, location,
iv. Compare the value calculated in above in (iii) with the incoming hash value and
v. If the two hash values match, this means that the message is coming from a trusted source; otherwise, it will be disregarded and dropped.

Since calculating the VTW window is cornerstone for this solution framework, we will focus on implementing the framework's first phase in this paper. The remainder of the paper is organized as follows. Section 2 provides an overview on the BSM message format, Section 3 describes how the simulation experiments are designed and executed. Sections 4 and 5 conclude the paper and outline the future work.

II. BASIC SAFETY MESSAGE (BSM)

By default, BSMs are periodically broadcasted to nearby vehicles carrying the safety messages. However, user custom information can be added to the BSM messages using custom fields. An example for custom info is a pull-over BSM message sent to a police vehicle to a violating vehicle. Messages that carry such sensitive info should be verified to ensure its legitimacy while protecting the identity of the driver or vehicle [3].

According to the SAE J2735 DSRC definition, BSM message consists of data elements (DEs) and data frames (DFs). The data frame consists of one or more data elements or other data frames. Furthermore, the BSM message consists of two sections such as shown in table 1. The first section is a required section of any BSM message and is known as Basic Vehicle State with a size of 39 Bytes. The second section of the BSM is optional and contains the Vehicle Safety Extensions and the Vehicle Status data frames. In general, vehicles periodically broadcast the first section of the BSM message only. However, in some events such as emergency braking, the BSM message can be further described by setting the corresponding event flag in the second part of the BSM, [1].

Table 1: Basic Safety Message (BSM)

BSM Data Item	Sequence	BSM Part	Type	Bytes
Message ID		I	Data Element	1
Message Count		I	Data Element	1
Temporary ID		I	Data Element	4
Time		I	Data Element	2
Latitude	PositionLocal3D	I	Data Element	4
Longitude		I	Data Element	4
Elevation		I	Data Element	2
Positioning Accuracy		I	Data Frame	4
Transmission & Speed	Motion	I	Data Frame	2
Heading		I	Data Element	2
Steering Wheel Angle		I	Data Element	1
Accelerations		I	Data Frame	7
Brake System Status	Control	I	Data Frame	2
Vehicle Size	Vehicle Basics	I	Data Frame	3

To keep track of the number of BSM messages sent out over time, the BSM message format contains a field called MsgCount. It is simply a number that serves the same purpose as sequence numbers in traditional networking protocols. Fig. 2 is developed by a 3D Simulink model. It displays the progression of the MsgCount field of two transmitting vehicles over time (one vehicle is pink, the other is yellow). Fig. 2 shows that two cars are traveling together from the right of the screen to the left, while a third one is traveling on the opposite direction is the receiving vehicle.

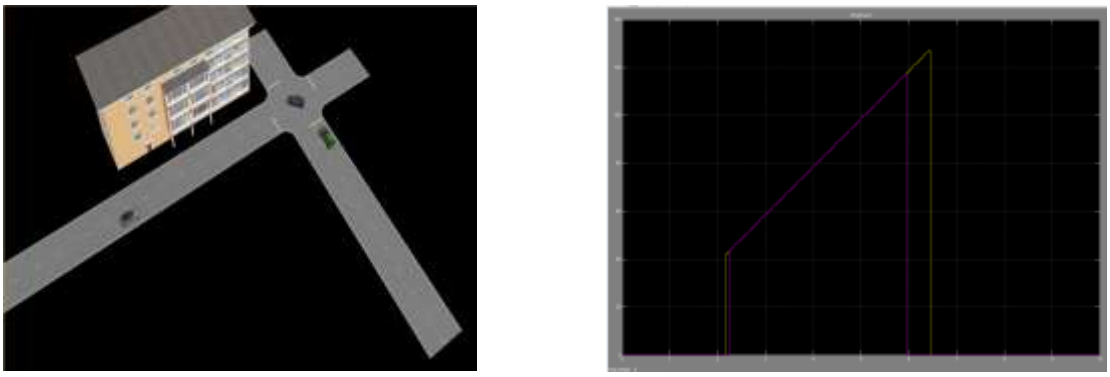


Figure 2. MsgCount Field Progression over Time for Two BSM Messages

III. EXPERIMENTAL ANALYSIS

We carried out multiple simulation experiments to calculate the visibility time between two testing vehicles. The visibility time represents the amount of time during which the transmitting vehicle was within the receiving vehicle's range. In the next sections, we'll introduce our simulation environment implemented by PreScan and the details of the conducted experiments to calculate the visibility time in different settings.

3.1 Prescan Simulation Engine

PreScan is a simulation development environment that is used to simulate realistic driving conditions and allow for testing users' algorithms and ADAS services in realistic simulation environment. It supports communications using various sensor technologies such as built-in vehicle camera, LIDAR, RADAR, and GPS in V2V and V2I architecture. PreScan supports three design paradigms: model-based controller design (MIL), real-time tests with software-in-the-loop (SIL) and hardware-in-the-loop (HIL) systems [4 and 5]. PreScan provides a GUI that enables us to build a simulation scenario and model sensors, while the Simulink and MATLAB interface allows us to add a control system to define the simulation building blocks to test the ADAS application or the user algorithm. There are four steps to build and run a simulation scenario such as described below and in Fig. 3.

a. Building the simulation scenario

A dedicated GUI is used to build and modify traffic scenarios using an existing database of road sections, trees, buildings, traffic signs, different vehicles such as cars, trucks and various weather conditions such as rain, snow and fog.

b. Modeling sensors

Various types of sensors such as radar, laser, camera, ultrasound, infrared, GPS and antennas for vehicle-to-X (V2X) communication can be added with various parameters to adjust to fit the simulation scenario.

c. Adding control system

A Matlab/Simulink interface enables us to control the vehicle movement algorithm as well as sensor fusion and adding any user custom module to perform a mathematical operation such as calculating the inter-vehicle distance in a simulation environment and exporting collected data to output files.

d. Running the simulation experiment

A 3D visualization viewer allows users to monitor the progress of the simulation experiment as well as analyze the obtained results.

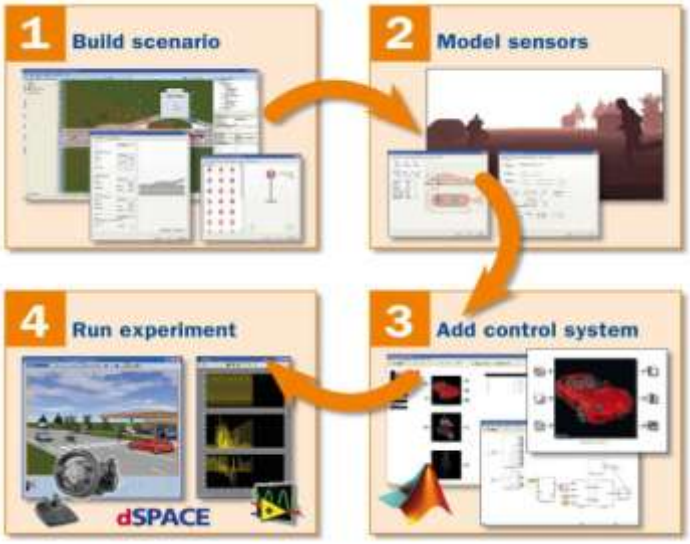


Figure 3. PreScan Simulation Engine Life Cycle [5]

3.2 Building Simulation Experiments

PreScan has been chosen to implement the simulation environment because it supports DSRC communications. We setup and ran PreScan and Simulink simulation experiments to determine the visibility time between two testing vehicles using different scenarios such as described in table 2, 3, 4 and 5:

Table 2: Control Experiment Parameters

Experiment1: Control Experiment Parameters					
Vehicle Size		**Vehicle Speed**		**Vehicle Trajectory**	
Sending Vehicle	**Receiving Vehicle**	**Sending Vehicle (m/s)**	**Receiving Vehicle (m/s)**	**Road Length(m)**	**Number of Lanes**
Sedan - Citreon C3 - Travels along westernmost northbound lane, turns left at intersection	Sedan - BMW Z3 - Travels along southernmost eastbound lane, turns left at intersection	15	15	120	2 opposite lanes

Table 3: Truck Experiment Parameters

Experiment 2: Truck Experiment Parameters					
Vehicle Size		Vehicle Speed		Vehicle Trajectory	
Sending Vehicle	**Receiving Vehicle**	**Sending Vehicle (m/s)**	**Receiving Vehicle (m/s)**	**Road Length(m)**	**Number of Lanes**
Truck/Nissan Cabstar - Travels along westernmost northbound lane, turns left at intersection	Sedan - BMW Z3 Travels along southernmost eastbound lane, turns left at intersection	15	15	120	2 opposite lanes

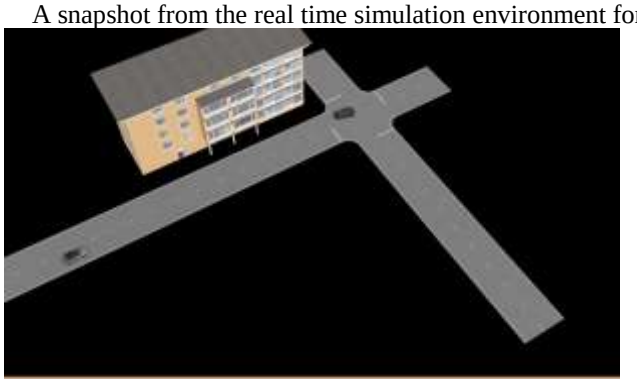
Table 4: Five Lanes Experiment Parameters

Experiment 3: Five Lanes Experiment Parameters					
Vehicle Size		Vehicle Speed		Vehicle Trajectory	
Sending Vehicle	**Receiving Vehicle**	**Sending Vehicle (m/s)**	**Receiving Vehicle (m/s)**	**Road Length(m)**	**Number of Lanes**
Sedan - Citreon C3 - Travels along westernmost northbound lane, turns left at intersection	Sedan - BMW Z3 - Travels along southernmost eastbound lane, turns left at intersection	15	15	120	5 opposite lanes

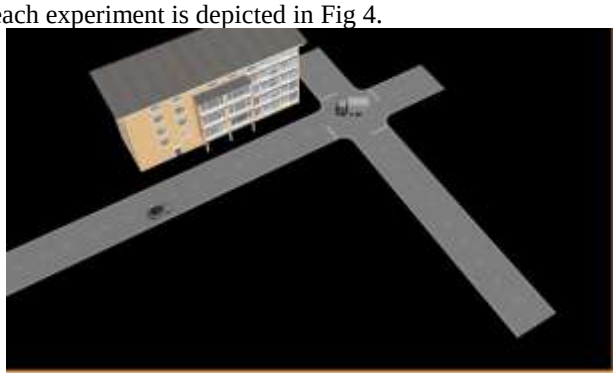
Table 5: 25 Meter per Sec Speed Experiment Parameters

Experiment 4: Twenty-Five MPS Experiment Parameters					
Vehicle Size		Vehicle Speed		Vehicle Trajectory	
Sending Vehicle	**Receiving Vehicle**	**Sending Vehicle (m/s)**	**Receiving Vehicle (m/s)**	**Road Length(m)**	**Number of Lanes**
Sedan - Citreon C3 - Travels along westernmost northbound lane, turns left at intersection	Sedan - BMW Z3 - Travels along southernmost eastbound lane, turns left at intersection	25	25	120	2 opposite lanes

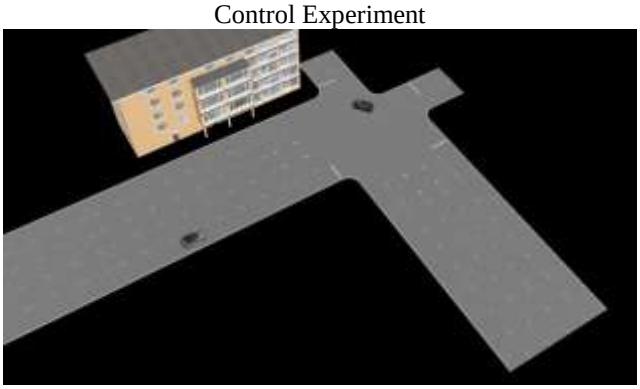
A snapshot from the real time simulation environment for each experiment is depicted in Fig 4.



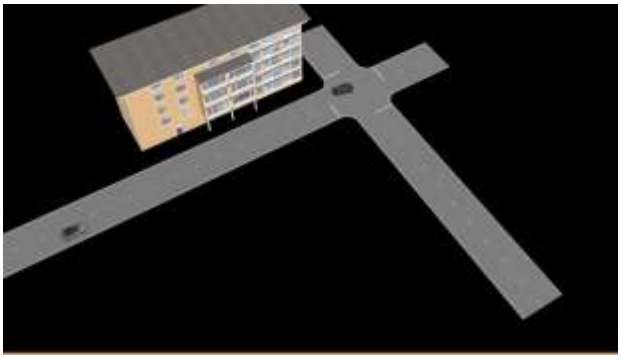
(a) Real Time View of the Simulation Environment For the Control Experiment



(b) Real Time View of the Simulation Environment For the Truck Experiment



(c) Real Time View of the Simulation Environment For the 5 Lanes Experiment



(d) Real Time View of the Simulation Environment For the 25 m/s Speed Experiment

Figure 4. PreScan Simulation Environment

The data collected from the simulation experiments are synchronous across all experiments, which means that the time samples for all experiments match up to that of the control experiment without error. Simulation results obtained are summarized in table 6.

Table 6: Simulation Results Summary

Experiment Type	First BSM Message Time (Sec)	Last BSM Message Time (Sec)	Visibility Time (Sec)
Control Experiment (1)	2.2	5.90	3.75
Truck Experiment (2)	2.2	5.95	3.80
Five Lanes Experiment (3)	2.2	6.20	3.95
25 MPS Speed Experiment (4)	1.80	4.25	2.45

We have observed that, as long as the vehicles speed is the same, the visibility time is almost the same such as shown below:

- Control experiment (2.2→ 5.90: 3.75Sec),
- Truck experiment (2.2→5.95: 3.80Sec),
- Five Lanes experiment (2.2→6.20: 3.95Sec),
- 25m/s experiment (1.80→4.25: 2.45Sec).

This means that the receiving vehicle is able to detect its peer vehicle at an average visibility time of *3.83 Sec*. In other words, the two vehicles will be able to hold an active authenticated communications session for 3.83 Sec starting from receiving the first BSM message. The only exception here is when the two vehicle are travelling with a high speed (25m/s), the visibility time = 2.45 Sec. Therefore, the vehicle that is traveling at a higher speed (25 m/s) was detectable for less time and they are able to hold a shorter V2V communications session. So, when vehicles travel at the same speed for instance at (15m/s), vehicles can predict for how far and for how long they will be engaged in an authenticated communication session. Assume that before a target vehicle is being within range of an attack vehicle, the attack vehicle will know neither its distance from the victim vehicle nor its victim's receiving range. Consequently, it will be very difficult for an attacker to successfully synchronize an attack to get a spoofed fake message through without any outside help because the target vehicle is sensitive not only to the contents of the BSM, but also to the behavior of the BSM's contents over time.

ACKNOWLEDGEMENTS

The authors would like to thank the TASS International Company in MI, USA for allowing us an opportunity to test and practice with their PreScan Simulation environment.

IV. CONCLUSION

In this paper, authors focused on predicting the authentication session duration based on estimating the visibility time between 2 vehicles in a V2V vehicle communication scenario. Authors introduced the visibility time window as a measure for the authentication session duration. Authors conducted simulation experiments using PreScan to study measure the visibility time between two vehicles in a V2V environment. The effect of several driving environment conditions have been studied and analyzed such as vehicle size, speed, and number of road lanes.

We found that it is fairly easy that vehicles can estimate how long their communication session can last for from receiving the first BSM message. Any messages arrived outside the visibility time window may be ignored

Future Work

Authors would like to investigate the implementation and evaluation of the second phase of the proposed framework.

REFERENCES

[1]. A. Svenson, G. Peredo, and L. Delgrossi (2015). Development of a Basic Safety Message for Tractor-Trailers for Vehicle-to-Vehicle Communications. 24th International Technical Conference on the Enhanced Safety of Vehicles (ESV), Gothenburg, Sweden

[2]. G. Corser, A. Arenas, and H. Fu (2016). Effect on vehicle safety of nonexistent or silenced basic safety messages. 2016 International Conference on Computing, Networking and Communications (ICNC), Kauai, Hawaii, USA.

[3]. J. Carter, and N. Paul (2015). Analysis of Vehicle-Based Security Operations. 24th International Technical Conference on the Enhanced Safety of Vehicles (ESV), Gothenburg, Sweden

[4]. M. Huang, Visibility and confidence estimation of an onboard-camera image for an intelligent vehicle. (2015), Accessed 2016
[5]. PreScan automotive simulation, https://www.tassinternational.com/prescan, Accesses 2016
[6]. C. Smith (2014). Car Hacker's Manual Paperback, Theia Labs