

SECURE MULTIMEDIA MESSAGING SERVICE IN VEHICULAR AD-HOC NETWORK

A Thesis

by

SATYA SRIDHAR KARANKI

Submitted to the College of Graduate Studies  
Texas A&M University-Kingsville  
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

December 2016

Major Subject: Computer Science

ProQuest Number:10255512

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



ProQuest 10255512

Published by ProQuest LLC (2017). Copyright of the Dissertation is held by the Author.

All rights reserved.

This work is protected against unauthorized copying under Title 17, United States Code  
Microform Edition © ProQuest LLC.

ProQuest LLC.  
789 East Eisenhower Parkway  
P.O. Box 1346  
Ann Arbor, MI 48106 – 1346

SMMV: SECURE MULTIMEDIA MESSAGING SERVICE IN VEHICULAR AD-HOC  
NETWORK

A Thesis

by

SATYA S. KARANKI

Approved as to style and content by:



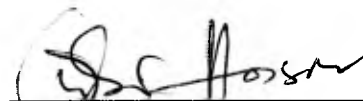
Mohammad Khan, Ph.D.  
(Committee Chairman)



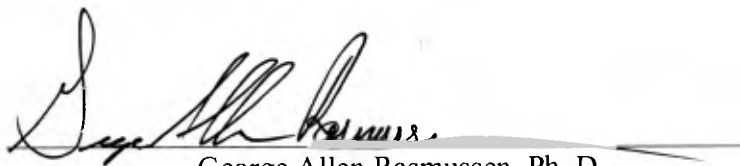
Rajab Chaloo, Ph.D., P.E.  
(Committee Co-Chairman and Chair  
of the Department)



David Hicks, Ph.D.  
(Member of Committee)



Gahangir Hossain, Ph.D.  
(Member of Committee)



George Allen Rasmussen, Ph. D.  
(Associate VP for Research and  
Dean of Graduate Studies)

December 2016

## ABSTRACT

### Secure Multimedia Messaging Service in Vehicular Ad-hoc Network

(December 2016)

Satya Sridhar Karanki, B.Tech, Nagarjuna University, M.S., University of Texas at San Antonio

Chairman of Advisory Committee: Dr. Mohammad Shoeb Saeed Khan

According to the National Highway Traffic Safety Administration (NHTSA) 38,300 people in the United States (U.S.) lost their lives in traffic accidents, and approximately 4 million people were injured in auto accidents in 2015. In economic terms, accidents cost approximately \$200 billion per year. An approach has been drafted to minimize auto accidents. The vehicular ad-hoc network enables communication between vehicles on the road in order to avoid accidents and solve traffic issues. This strategy enables communication between vehicles, when a major accident or obstruction occurs. The secondary purpose of the ad-hoc network is to enable nonsafety applications that can provide infotainment to travelers. The primary objective of this thesis was to build an Android-based application that utilizes WiMAX IEEE 802.16 network protocols to send a multimedia message securely from one vehicle to another vehicle. The proposed protocol combines advanced encryption standards and secure hash algorithms to protect the message. Thus, it enables the driver to send multimedia messages securely to other drivers using voice commands.

## TABLE OF CONTENTS

	Page
CHAPTER I. INTRODUCTION.....	1
CHAPTER II. LITERATURE REVIEW .....	5
CHAPTER III. RESEARCH APPROACH .....	10
3.1    ADVANCED ENCRYPTION STANDARD .....	10
3.2    SECURE HASH ALGORITHM .....	13
3.3    SMMV PROGRAM DESIGN .....	16
3.4    ASSUMPTIONS .....	18
3.5    PROBLEM STATEMENT AND PROPOSED PROTOCOL.....	18
3.6    SMMV ENCRYPTING AND DECRYPTING ALGORITHM .....	19
CHAPTER IV. APPLICATION DEVELOPMENT .....	24
4.1    USER INTERFACE MANAGER .....	24
4.2    DATA MANAGER .....	24
4.3    MESSAGING MANAGER .....	25
4.4    CRYPTIC MANAGER.....	25
4.5    SERVER MANAGER .....	25
4.6    NETWORKING MANAGER .....	26
4.7    USING EVENT BUS.....	27
4.8    CLASS DIAGRAM .....	28

	Page
4.9 SEQUENCE DIAGRAM.....	29
CHAPTER V. SECURITY PERFORMANCE EVALUATION .....	31
CHAPTER VI. PERFORMANCE ANALYSIS OF MESSAGE TRANSMISSION .....	35
CHAPTER VII. PERFORMANCE ANALYSIS OF ROAD SIDE UNIT AND WORLDWIDE INTEROPERABILITY FOR MICROWAVE ACCESS.....	37
7.1 SIMULATION RESULTS .....	37
7.1.1 VEHICULAR SPEED VS. PACKET LOSS RATIO.....	38
7.1.2 VEHICULAR SPEED VS. END-TO-END DELAY .....	39
CHAPTER VIII. CONCLUSION.....	41
REFERENCES .....	42
VITA.....	47

## LIST OF TABLES

	Page
Table 1: Wireless Technologies Comparison. ....	6
Table 2: Showing time taken for encryption and decryption in computational systems .....	32
Table 3: Showing time taken for encryption and decryption with variable data sizes. ....	33

## LIST OF FIGURES

	Page
Figure 1: Various Wireless communications in Vehicular ad-hoc network.....	2
Figure 2: Advanced Encryption Standard Encryption Process.....	11
Figure 3: Shift Rows.....	12
Figure 4: Advanced Encryption Standard Decryption Process.....	13
Figure 5: Overview of Secured message communication between vehicles using Worldwide interoperability for Microwave Access.....	18
Figure 6: Flow chart showing the message flow from Sender side and Recipient side.....	21
Figure 7: SMMV communication between vehicles .....	23
Figure 8: Functional Block diagram showing interactions of modules in the Secure Multimedia Message Application.....	26
Figure 9: Flow of Events interacting with Event Bus.....	27
Figure 10: Class Diagram .....	28
Figure 11: Sequence Diagram showing the events flow when user sends an Encrypted Message.....	29
Figure 12: Sequence Diagram showing the events flow when a recipient receives an Encrypted Message.....	29
Figure 13: Time taken for Encryption and Decryption with increase in Message size .....	34
Figure 14: SMMV protocol vs. Certificate-Based Authentication protocol.....	35
Figure 15: Impact of vehicular speed in percentage of packet loss .....	38
Figure 16: Impact of vehicular speed in percentage of packet loss on highway .....	39
Figure 17: Impact of vehicular speed on message delivery .....	39
Figure 18: Impact of vehicular speed on message delivery on highway .....	40



## CHAPTER I

### INTRODUCTION

World Health Organization statistics show that over 1 million people died globally and 20-50 million people were injured or disabled in road accidents in 2013 [1]. According to the American National Safety Council, 38,300 people lost their lives in traffic accidents, and approximately 4 million people were injured in auto-related collisions in 2015. These statistics represent a 14% increase in loss of life and accidents compared to data in 2013 and 2014 [2]. In economic terms, accidents in the United States (U.S.) cost almost \$200 billion per year [3].

Technology can play a vital role in preventing road accidents and ensuring safety. Vehicular early warning systems gained high priority in recent years and are used to detect the road conditions or alert drivers about accidents providing safe transportation. Developments in the field of wireless communication and automobile industries made vehicular ad-hoc network (VANET) into one of the most promising fields in academic and research sectors. Due to its unique characteristics, such as dynamic topology and predictable mobility, VANET has become more popular as a methodology for preventing accidents [4].

In VANET, each vehicle acts as a wireless node or router and connects to another vehicle in order to create a broad range of the mobile vehicular network. Any number of vehicles can join or leave the network at any time. This wireless network enables comprehensive communication between vehicles [5].

Communication between vehicles include three types: vehicle to vehicle communication (V2V), vehicle to road side unit (RSU), and vehicle to infrastructure, such as WiMAX towers. When a major accident or obstruction occurs, this network helps in terms of communicating the requisite information between the vehicles.

Road side units (RSUs) use wireless access in vehicular environment – IEEE 802.11p (WAVE) which is standard protocol for communication between vehicles and other RSU. These are stationary objects placed on the road side. They operate at 5.9 GHz and support short-range communications. These include devices that have the ability to exchange necessary information from high-speed vehicles such as traffic and road hazards [6].

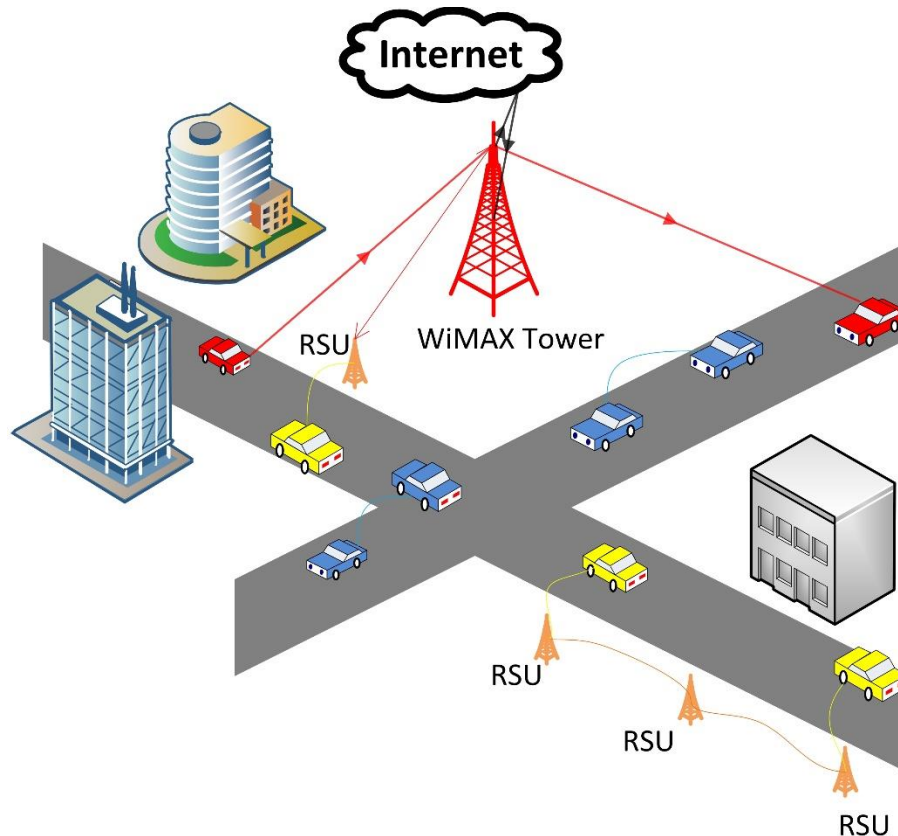


Figure 1: Various Wireless communications in VANET

Worldwide Interoperability for Microwave Access (WiMAX), operates at IEEE 802.16. All the IEEE standards under 802.16 come under the heading of the WiMAX family. It exhibits a longer range (approximately 50 kms), therefore, supporting long-range communications within a bandwidth of 2.5 GHz. It also supports greater download and upload rates within high-speed vehicle environments [7]. To improve the performance metrics, a combination of 802.16 and 802.11p has also been introduced [8].

The primary concern of VANET is the safety of the passengers in the vehicles. However, numerous applications have been proposed which include value added nonsafety features. Based on the type, VANET applications are broadly divided into safety and nonsafety applications. Applications that convey critical safety information, including but not limited to road conditions, accidents, and traffic related information through the data communication between vehicles, RSUs or roadside infrastructures, comes under the heading of safety applications [8].

The secondary purpose of VANET is to enable the multifarious nonsafety applications that can provide infotainment to travelers [9]. These are entertainment-based applications that provide online connectivity, video or audio streaming, and message communications. Google's Android Auto [10] and Apple's Apple Car Play [11] are examples of dashboard projections that provide information and entertainment through mobile devices to the vehicle's travelers. These systems are completely dependent on mobile operating systems.

The Android compatibility definition document released on October 2015 refers to a vehicle dashboard display, running on the Android operating system called Android automotive. Rather than using previous versions of Android auto, which required a mobile to function, Android automotive works independently in order to provide a system with interdisciplinary infotainment functionality [12].

The concept of Android automotive is in the budding stage; focus on using Android applications for message communications between vehicles is a fairly new approach. This thesis focuses on the explication of developing an Android application protocol deemed secure multimedia messaging service in vehicular ad-hoc network (SMMV) that can be used for personal message communications between vehicles using WiMAX network securely.

Considerable research has been done on message communication between vehicles using road side unit (RSU) [6], [13], [14], [15], [16], but little extant research has been conducted on

using WiMAX towers. Road side units and vehicles have to maintain secure keys or certificates to authenticate the message transmission from each unit or vehicle. This requires storage capacity in RSUs and vehicles which is restricted [6]. As the number of vehicles and RSU increase, the number of keys and certificates needed also increases excluding the maintenance cost of keys or certificates.

The advantage of using the Android application with WiMAX is that every time the application connects to the server it authenticates itself. Therefore, there is no need for maintaining other vehicle certificates or keys to authenticate for message communication. The Android application uses a secure communication channel (HTTPS) to connect to the application and the server. Additionally, the message is encrypted end-to-end with AES for message authenticity and uses SHA-256 to maintain its integrity. For efficient transmission, ad-hoc on-demand multipath distance vector routing (AOMDV) protocol is used.

## CHAPTER II

### LITERATURE REVIEW

Vehicular ad-hoc network is becoming one of the most promising emergent technologies as it helps in preventing accidents by alerting the drivers about the incidents or conditions that are ahead on the route while also notifying law enforcement or medical teams about the incidents so that they can prevent further loss. A good amount of research conducted, not only in vehicular safety but in also providing infotainment solutions, has reinforced the robustness of the VANET approach.

Vehicular ad-hoc network exhibits numerous choices in wireless technologies to communicate with other vehicles. These are divided into three categories: long, medium and short ranges. Short-range technologies include infrared, bluetooth, zigbee, and ultra-wideband (UWB). They operate over a distance of 10 to 70 m and the data transfer rate is from 250 Kbps to 4 Mbps. Because of the low transfer rate and short range, these methods are used in certain applications such as toll collections, parking spot locators, and parking availability notifications.

Medium-range wireless technologies include Wi-Fi, dedicated short range communication (DSRC) or wireless access in vehicular environment (WAVE), continuous air-interface long and medium range (CALM). Depending on the wireless standard Wi-Fi operation, the range of service is between 100 m to 250 m at a speed of 11–100 Mbps. Dedicated short range communication (DSRC) comes under IEEE 802.11p wireless technology. They operate at 5.9 GHz frequency and at a speed of 27 Mbps over a range of 1 km, while CALM operates at 5 GHz frequency with an operational range of approximately 10 km at a speed of 6 Mbps. Among these options, DSRC signal interference is mitigated with minimal maintenance cost. Though the operational range is slight compared to CALM, the data transfer rate is significantly higher and

meets the VANET's requirement for enhanced dynamic topology and high mobility. Applications related to road safety use DSRC for its secure data transmission and reliability.

Table 1: Wireless Technologies Comparison [7].

Wireless Standard	Frequency (f) in GHz	Data Transfer Rate	Max Signal Coverage $\approx$	Signal Interference	Maintenance
Cellular Systems	Operator Dependent	$\approx$ 384 Kbps -129 Mbps	50 km	Low	Difficult
WiMAX 802.16m	2.3/2.5/3.5	$\approx$ 75- 300 Mbps	50 km	High	Difficult
MBWA 802.20	3.5	$\approx$ 4.5 Mbps	15 km	High	Easy
Wi-Fi 802.11a	5.1/5.8	$\approx$ 54 Mbps	100 m	Low	Easy
Wi-Fi 802.11b	2.4	$\approx$ 11 Mbps	100 m	High	Easy
Wi-Fi 802.11g	2.4	$\approx$ 54 Mbps	140 m	High	Easy
Wi-Fi 802.11n	2.4/5	$\approx$ 100 Mbps	250 m	High	Easy
DSRC 802.11p	5.8/5.9	$\approx$ 27 Mbps	1 km	Low	Easy
CALM M5	5	$\approx$ 6 Mbps	10 km	High	Difficult
Infrared	300 GHz - 400 THz	$\approx$ 115 Kbps – 4 Mbps	100 m	Low	Easy

Long-range wireless technologies include cellular, WiMAX, and mobile broadband wireless access (MBWA). The frequency range of cellular networks depends on the operator, and data transmission rates ranging from 384 kbps to 129 Mbps while the maximum coverage

area is 50 km. The WiMAX frequency range is 2.5 GHz with a data transfer rate ranging from 75-300 Mbps and maximum coverage area of 50 km. On the other hand, the average microwave frequency range is 3.5 GHz with a data transfer rate of 4.5 Mbps and maximum coverage area of 15 km. Though cellular and WiMAX are the closest competitors, WiMAX was selected for this project based on the transmission speed [7]. Table 1 gives a comparative analysis of various wireless communication technologies discussed above.

Anwer and Guy, from research article [7], stated that, since WiMAX provides bandwidth higher than LTE (2.5 GHz), it increases the throughput.

In 2011, the IEEE standard association approved IEEE 802.16m (WiMAX) [17]. Among emerging broadband technologies that provide high-speed Internet, WiMAX is one of the best compatible wireless technologies for utilizing VANET. Vinoth and Manoreya, in their recent publication, mentioned that WiMAX aimed to provide download rates of 100 mb/sec at a speed of 340 km/hr [18].

As discussed earlier, VANET is a group of mobile nodes moving at variable speeds; each node acting as a host and router that can go out of the network or join in a network. Since these are dynamic topologies, one of the key challenges is to design a dynamic routing protocol. Johnson and Maltz [19] proposed a protocol called dynamic source routing (DSR) that adapts to routing changes with little or no overheads. Perkins and Royer [20] introduced a novel algorithm called ad-hoc on-demand distance vector routing (AODV). It considers each vehicle as a router and attains the route on demand. Whenever this protocol initiated, a request sent to the destination through a network and subsequently waiting for a reply, depending on latency and overhead, this may take a longer time for large networks. If it failed to discover a route because of troubleshooting issues, a new request is initiated to find the destination, which may take an even longer time. To overcome these hurdles, Marina and Das [21] have proposed the ad-hoc on-

demand multipath distance vector routing (AOMDV) protocol. Instead of utilizing a single path suggested by AODV, AOMDV protocol discovers multiple paths originating from the source to the destination, and the best route is selected among them. Moreover, it has to invoke a new path only when all the previous paths fail.

The performance of AODV, AOMDV, and DSR routing protocols have been analyzed recently by Dorge and Dorle [22]. Their design is based on a WiMAX network with multiple inputs multiple outputs (MIMO) and adaptive modulation coding (ADC) techniques. They have used an NS2 simulator to create realistic scenarios such as high-speed highway environments, variable speed vehicle environments, and city environments. They concluded that AOMDV routing protocol is better for WiMAX-based VANET systems than the other two protocols based on the quality of service, maximum throughput, and packet delivery ratio.

Most of the research in transmitting secure messages is based on RSUs utilization of IEEE 820.11p such as [6], [13], [14], [15], [16] . However, scarce research has been initiated pertaining to communicating personal messages based on Android-based applications through VANET between vehicles. Since VANET uses wireless technologies, care has been taken to minimize the risk of security threats.

In 2005, Google acquired Android operating system. It is being used as a mobile operating system for cell phones and tablets. The advances in Android technology have grown in such a way that Android open source projects, in the latest upgrades, embraced a variety of devices such as Android television, watches, handheld devices, and automotive applications [12].

As discussed earlier, the proposed system uses Android automotive to send secure multimedia messages from one vehicle to the other. Android features multilayer security that ensures the safety of the users, network, applications, data, and the device [23]. This protocol uses the https connections to connect the server in order to create secure web traffic [24].



Zou et al. [25], described the WiMAX protocol stack and security sublayers. There are two layers in the protocol stack, medium access control (MAC) layer and physical layer. Medium access control layer constitutes three more sublayers: service specific convergence sublayer, common part sublayer, and security sublayer. All the security-related issues are managed under security sublayer. Additionally, it is responsible for authentication, authorization, and encryption. MAC layer is well-protected while physical layer is vulnerable to security threats such as eavesdropping, jamming, and scrambling attacks. Eavesdropping constitutes secretly listening to someone's conversation. Jamming reduces the channel capacity while scrambling is similar to jamming but with short bursts targeted at specific intervals. Either way, the message transmitted from one person to another is not private anymore.

## CHAPTER III

### RESEARCH APPROACH

The proposed system is a Java-based Android application that combines AES encryption algorithm with secure hash functions (SHA -256). However, the messages transfer from one user to another user using secure socket layer (SSL, https), and additional securities such as AES and SHA-256 are used to provide end-to-end encryption, ensuring only end users can read it.

#### 3.1 ADVANCED ENCRYPTION STANDARD

In 2001, the National Institute of Standards and Technology (NIST) established a specification for the encryption of electronic data called advanced encryption standard (AES) [26]. Daemen and Rijmen from Belgium developed this AES algorithm based on Rijndael cipher. Advanced encryption standard utilizes a symmetric key algorithm, meaning that it uses the same key for encryption and decryption of the messages. The design principle employed in AES is substitution-permutation network, which is a combination of permutation and substitution. Its block size is fixed to 128 bits, and the key size is 128/192/256 bits' length which is selected based on the encryption strength [27]. Advanced encryption standard is an open-source algorithm. The American government has dubbed AES as the standard algorithm for encryption. It can be implemented in hardware and software, as well as in confined environments (i.e., smart card) providing good defense techniques on various attacks.

#### Encryption Process

There are four steps involved in AES encryption processes. Based on the key size, these steps are repeated. One round involves all four steps and the 128-bit key has 10 rounds.

1. Sub bytes
2. Shift rows

3. Mix columns
4. Add round key

### Key Expansion

In the key expansion step, 16 bytes in 128-bit key are arranged in a 4 X 4 matrix. The first four words make the first column ( $w_0$ ), and similarly, the next column will make  $w_1, w_2, w_3...$  This is expanded to  $w_{44}$  (176 bytes) using a method called key schedule. Therefore, there will be 44 4-bit words.

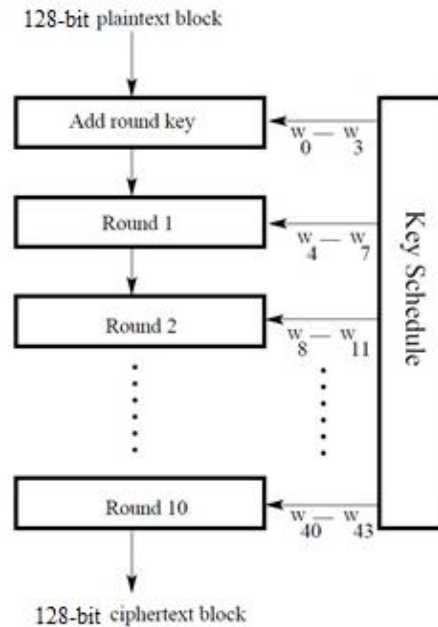


Figure 2: Advanced encryption standard Encryption Process

Advanced encryption standard is a block cipher. It encrypts the text or file in blocks of 128 bits. If the size is less than 128 bits, it will be padded during encryption. Before starting the encryption process, the input is divided into 128-bit blocks. First, 4 bytes from key expansion are XORed with the input, and the remaining bytes are XORed in each round to the input as shown in Figure 2.

## Sub bytes

In this step, each byte is substituted by another byte concerning a 256-element, 16 X 16 table. Each byte from the row is converted to hexadecimal value  $B_{ij}$ . From the table, the value of  $i^{\text{th}}$  row and  $j^{\text{th}}$  column is selected and replaced. Similarly, all bytes are substituted in the matrix.

## Shift Rows

In this step, the bytes are scrambled inside the 128-bit block. In the 4 X 4 matrix, the first row is undisturbed, and the first letter from the second row is shifted to the right. Then, two letters from the third row are moved to the right and, finally, the first three letters from the fourth row are shifted to the right.

$$\begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} \Longrightarrow \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,1} & s_{1,2} & s_{1,3} & s_{1,0} \\ s_{2,2} & s_{2,3} & s_{2,0} & s_{2,1} \\ s_{3,3} & s_{3,0} & s_{3,1} & s_{3,2} \end{bmatrix}$$

Figure 3: Shift Rows

## Mix Columns

In this step, each byte is multiplied by 2 plus, (XORed) 3 times the next byte plus, the third and fourth byte. For example, below the first-row elements as  $R_{0,0}$ ,  $R_{0,1}$ ,  $R_{0,2}$ ,  $R_{0,3}$ .

$$R_{0,0} = 2X R_{0,0} + 3X R_{0,1} + R_{0,2} + R_{0,3}$$

$$R_{0,1} = 2X R_{0,1} + 3X R_{0,2} + R_{0,3} + R_{0,0}$$

$$R_{0,2} = 2X R_{0,2} + 3X R_{0,3} + R_{0,0} + R_{0,1}$$

$$R_{0,3} = 2X R_{0,3} + 3X R_{0,0} + R_{0,1} + R_{0,2}$$

Similarly, all the elements are calculated. Additions in the above example are considered as XORed. Bytes are moved circularly during the calculation process.

## Add Round Key

Keys derived from the key expansion are used in this step. The resulting matrix from mix column step is XORed with the key  $w_4-w_7$  in the first step. The remaining keys are applied in the next nine steps.

In the last round (10th), the mix column step was skipped. Similarly, all the plain text was encrypted. Finally, a ciphertext was created that can only be decrypted using the correct key. During decryption, this process will move in the opposite direction. The steps include inverse shift rows, inverse substitute bytes, add round key and inverse mix columns [28].

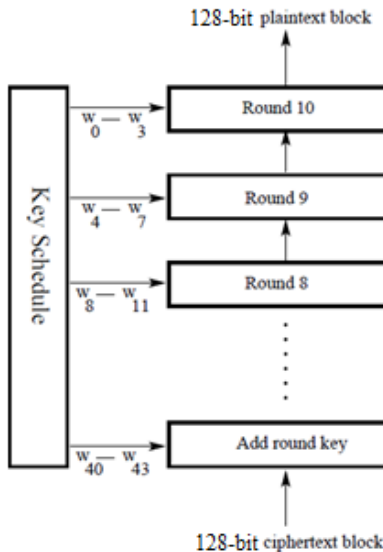


Figure 4: Advanced encryption standard Decryption Process

Though there are many other encryption tools, AES has proven its robustness in symmetric key encryption. Also, the National Security Agency (NSA) uses AES for encrypting top-secret classified information [26].

## 3.2 SECURE HASH ALGORITHM

Secure hash algorithm-256 is one of the six hash functions from the SHA-2 family. The National Institute of Standards and Technology (NIST), with the help of the National Security

Agency (NSA), developed the cryptographic hash function of SHA in 1993 and named it as SHA-0 [29]. Advances in computer technology helped Wang et al. to find a collision attack with 239 complexities within a practical range [30]. Fortunately, the NSA upgraded the hash function, SHA-1, making it difficult to break with a complexity of  $2^{63}$  computations. Though this is theoretically unhackable, no one has claimed to break this hash function [31]. At a later time, the NIST published the most sophisticated hash algorithm SHA-2 in 2001 with output ranges from 224 to 512 bit, termed SHA-224, 256, 384 and 512.

### **Hashing Process**

Digital data, such as a file or string, are converted into hash values using mathematical operations in cryptographic hash functions. In SHA-256, the hashing process consists of two steps padding the message and hash computation. The result of this process is a 256-bit length digested message.

### **Padding:**

Secure hash algorithm-256 processes the message in 512-bit blocks. Therefore, if the message (m) size is less than 512 bits (l), it is padded with zeros (0)(k) and one (1) is added to the last in order to make it  $448 \bmod 512 (l+k+1)$ . Then, each 512-bit block is parsed into 16 32-bit blocks ( $M^1, M^2, \dots, M^N$ ), which will be further processed in hash computation.

### **Hash Computation**

During hash computation, basic operations such as AND, XOR, and OR are used. A set of initial hash values are utilized in the hash calculation.

$$H_1^{(0)} = 6a09e667$$

$$H_2^{(0)} = bb67ae85$$

$$H_3^{(0)} = 3c6ef372$$

$$H_4^{(0)} = a54ff53a$$

$$H_5^{(0)} = 510e527f$$

$$H_6^{(0)} = 9b05688c$$

$$H_7^{(0)} = 1f83d9ab$$

$$H_8^{(0)} = 5be0cd19$$

The blocks that are parsed in the above step are processed one at a time as shown below.

“For t = 1 to N

$$(a, b, c, d, e, f, g, h) = (H_1(t-1), H_2(t-1), H_3(t-1), H_4(t-1), H_5(t-1), H_6(t-1), H_7(t-1), H_8(t-1))$$

Complete 64 rounds consisting of

$$T_1 = h + \Sigma_1(e) + Ch(e, f, g) + Ki + Wi$$

$$T_2 = \Sigma_0(a) + Maj(a, b, c)$$

$$h = g$$

$$g = f$$

$$f = e$$

$$e = d + T_1$$

$$d = c$$

$$c = b$$

$$b = a$$

$$a = T_1 + T_2$$

Compute the new value of  $H_j^{(t)}$

$$H_1(t) = H_1(t-1) + a$$

$$H_2(t) = H_2(t-1) + b$$

$$H_3(t) = H_3(t-1) + c$$

$$H_4(t) = H_4(t-1) + d$$

$$H5(t) = H5(t-1) + e$$

$$H6(t) = H6(t-1) + f$$

$$H7(t) = H7(t-1) + g$$

$$H8(t) = H8(t-1) + h$$

The hash of the message is the concatenation of the variables  $H_i^N$  after the last block process

$$H = H1(N) \parallel H2(N) \parallel H3(N) \parallel H4(N) \parallel H5(N) \parallel H6(N) \parallel H7(N) \parallel H8(N).$$

The output compared with the known and expected hash values determine the integrity of the file or string. Hash functions, such as SHA-256, are used to verify the data integrity during data transmission through a network” [32].

### 3.3 SMMV PROGRAM DESIGN

Creating a robust security environment with an extensive scale of attacks is the toughest challenge. The proposed framework must be constructed in such a way that a client can send the message to another client without security breaches. The security requirements are stand-alone and regarded as building blocks. The primary goal of the project was to achieve the following objectives.

**Authentication:** The Android application has to authenticate the end user and the vehicle. The application must respond only to the legitimate users. Since every user must have a Google account to use an Android application, this objective can be achieved effortlessly. The only restriction is if a person wants to send a message to friends, the other user must be on friend's list.

**Confidentiality:** This application has to protect the privacy of the message content. Advanced encryption standard protects the confidentiality of the message. Since the message is encrypted



with a random key, it is impossible to decrypt the message without the correct key. The recipient only can see the end message. A 128-bit length key is used for encryption in this project.

Encryption can be of two types: symmetric and asymmetric. In the symmetric approach, the same key is used to encrypt and decrypt the message while two keys, private and public, are used in an asymmetric approach. Since the asymmetric approach is not feasible in this project, the symmetric method was selected.

A common method used by hackers to decrypt the encrypted message is through a brute force attack. In this technique, all possible combinations are tested to decrypt the message. A 128-bit key has  $2^{128}$  combinations, which is practically impossible to crack through a brute force attack.

**Integrity:** The message integrity is a primary concern. Secure hashing algorithms protect message integrity. It is impossible to reverse the hashed message but it can compare the original encrypted message with received encrypted data in order to verify the integrity of the message.

**Minimal packet loss ratio:** Successful message transmission is one of the primary objectives that must be achieved. Ad-hoc on-demand multipath distance vector routing (AOMDV) protocol is used to find the best route to deliver the message. It delivers 88% of messages on a high-speed highway environment, while the ratio increased to 90% at variable speed vehicle environments. However, the ratio dropped to 83% within a city environment network [22]. In either of the cases, AOMDV has a maximum packet delivery ratio that is higher than AODV and DSR.

This research focused on developing an application to send an experimental MMS or a text message between two vehicles while making sure that the application maintained the above objectives.

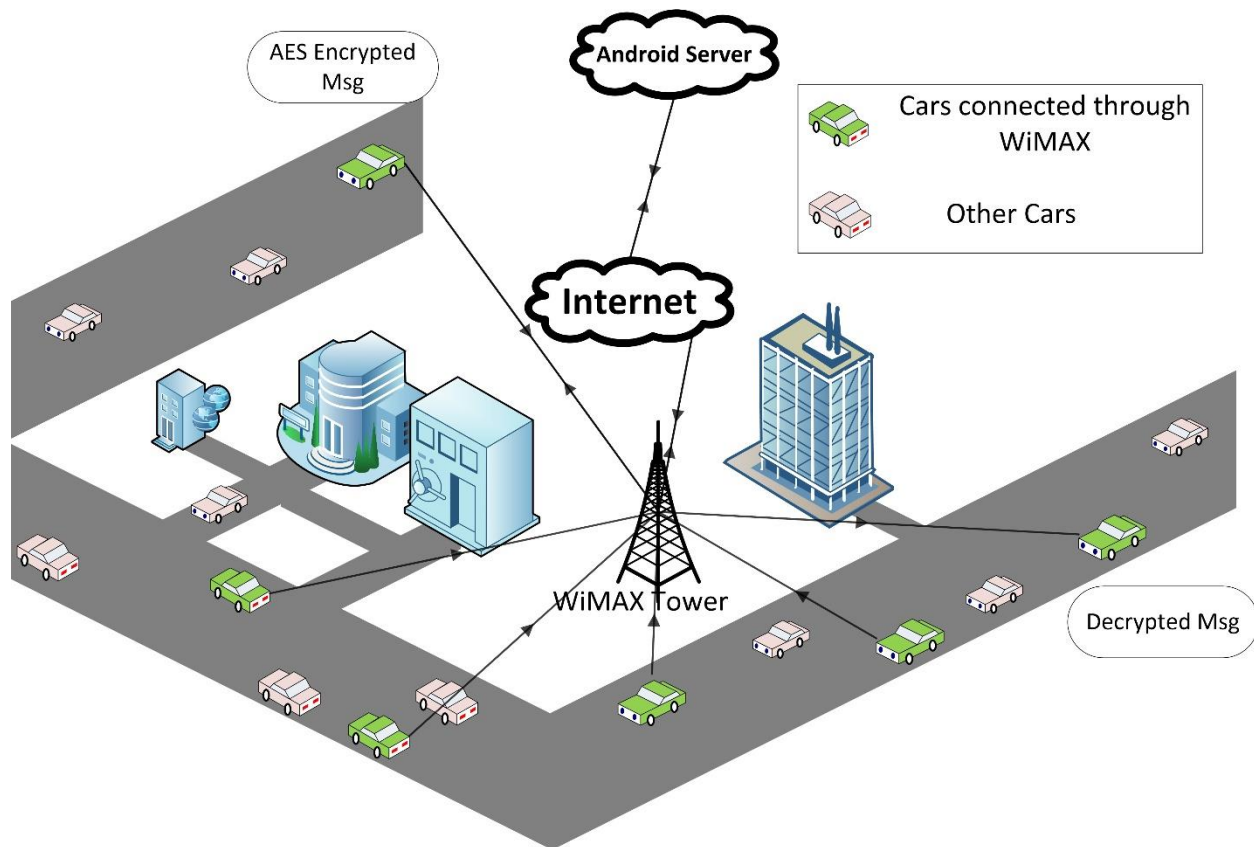


Figure 5: Overview of Secured message communication between vehicles using Worldwide interoperability for microwave access

### 3.4 ASSUMPTIONS

This thesis assumes that experimental vehicles equipped with Android automotive are functional and operational. The WiMAX IEEE 802.16 network is available for wireless communications between vehicles. All vehicles are capable of transmitting messages through WiMAX using AOMDV routing protocol.

### 3.5 PROBLEM STATEMENT AND PROPOSED PROTOCOL

As discussed in the literature review, limited extant of research on building Android applications that are used to send messages from one vehicle to another vehicle. However, some

research has been done on utilizing AES encryption to encrypt messages. A few prominent researchers have proposed a new secure schema for VANET that uses AES for encrypting the message [33], [34]. Since the authentication is based on the certificate-based public key, it creates an overhead and can delay the transmission [35]. Each vehicle has to maintain the root certificates or keys used by other vehicles, which may also consume the limited memory of vehicles.

The proposed Android application can overcome the certificate-related overheads. For instance, each vehicle will maintain only one certificate that is authenticated at the server level which will not create overheads during message transmission. Users must register with this application to use the service in the registration process, the application identifies the vehicle by its vehicle identification number (VIN) and creates a unique ID that uses internal operations. The application will have access to the address book of the user. In the address book, a friend can be added or deleted and the friends list can be sorted.

### **3.6 SMMV ENCRYPTING AND DECRYPTING ALGORITHM**

Secure multimedia messaging service in vehicular ad-hoc network is an Android application, every user must register to use it. During the registration process, vehicle identification number (VIN) and state registration information are collected to authenticate the user. Only the vehicle registered user can register to this application. Once the registration process is complete, this application will filter the Google contacts and displays the application users.

The sender creates his personal message  $M$  (converted to text  $T$  using Java API) and gives a voice command to send to a friend  $F$ . The application will query whether  $F$  is on the friends list. If present, then it goes to the next step or shows an error message, friend is not on the

list. A random key  $K$  is generated. In the next step, using the random key, the AES cryptographic algorithm encrypts the message  $E$ . Later, SHA-256 digests the encrypted message into a hashed message  $H(E)$ . Then, a random number  $R$  is generated between 0-49. Based on the random number, the encrypted message is subsequently divided into two parts as shown in equation 1.

$$H(E)/R1 + H(E)/R2 \quad (1)$$

The ad-hoc on-demand multipath distance vector Routing (AOMDV) determines the best route and sends the encrypted message using the WiMAX network. The bigger portion,  $H(E)/R1$ , of the message directly goes to the recipient while the smaller part, the hashed message and key, are stored in the server as shown in equation 2.

$$[H(E)/R2 + H(E) + K] \quad (2)$$

When the recipient receives the message  $H(E)/R1$ , the application collects the sender information and global positioning of the vehicle and sends it to the server. Using this information, the server sends the other part of the message, key and hashed message, to the recipient as shown in equation 2.

The application will concatenate the encrypted message, digest it using SHA-256 and compare it with the received hashed message  $H(E)r$  with the sender's hashed message  $H(E)s$  as shown in equation 3.

$$H(E)r = H(E)s \quad (3)$$

When both the hashed messages are the same, the application decrypts the message and shows the original message.

Figure 6 describes the flow of events when a user sends a message to the recipient, and the recipient opens the message.

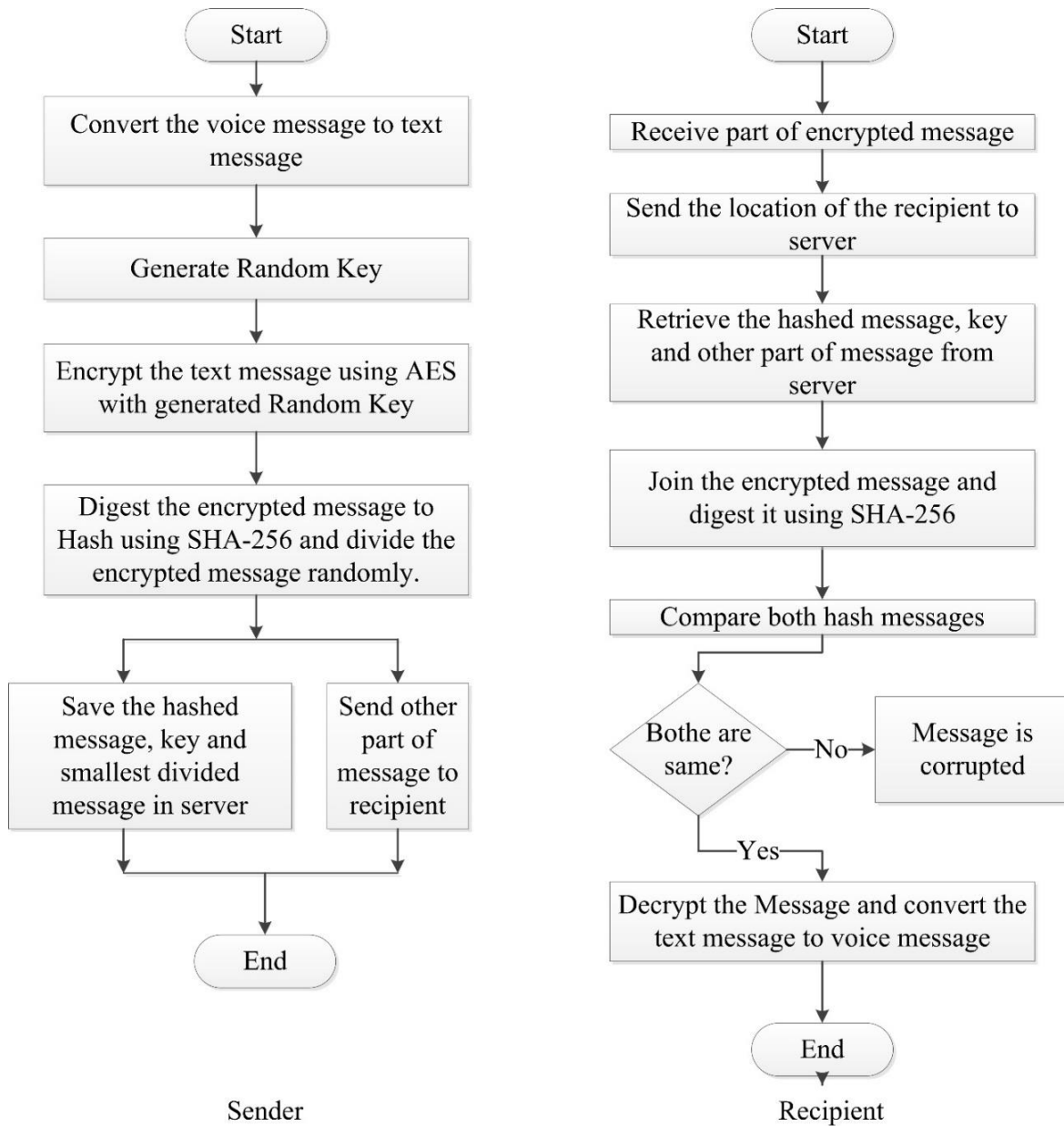


Figure 6: Flowchart showing the message flow from Sender side and Recipient side

## Algorithm 1

### Sender side

Step 1	if (Friend <b>F</b> contains in friends list) (True) { ... } else { return 'Friend not in the Friend's List' }
Step 2	Input <b>M</b> = received voice message if ( <b>M</b> translated to Text <b>T</b> ) (True) { ... } else { return 'Error' }
Step 3	<b>K</b> = Random Key
Step 4	Invoke AES Encryption with input <b>MT</b> BS (Bytes Substitution) SR (Shift Rows) MC (Mix Columns) ARK (Add Round Key) Output --> <b>E</b>
Step 5	Invoke SHA 256 Digest with input <b>E</b> PB (Padding Bits) HC (Hash Computation) Output --> <b>H(E)</b>
Step 6	Generate a Random number between 0-49 <b>R</b> .
Step 7	Divide the Encrypted message into two <b>H(E)/R1 + H(E)/R2</b>
Step 8	<b>B</b> = Best Route
Step 9	Send [ <b>H(E)/R2 + H(E) + K</b> ] to the server using <b>B</b> .
Step 10	Send <b>H(E)/R1</b> to the receiver using <b>B</b>

## Algorithm 2

### Recipient side

Step 1	Receive Input: <b>H(E)/R1</b>
Step 2	Invoke server to fetch <b>[H(E)/R2 + H(E) + K]</b>
Step 3	Complete the encrypted message and digest using SHA 256 <b>H(E)r</b>
Step 4	if ( <b>H(E)r = H(E)s</b> ) (True) { ... } Output: Decrypted Message else { return 'Message Corrupted' }

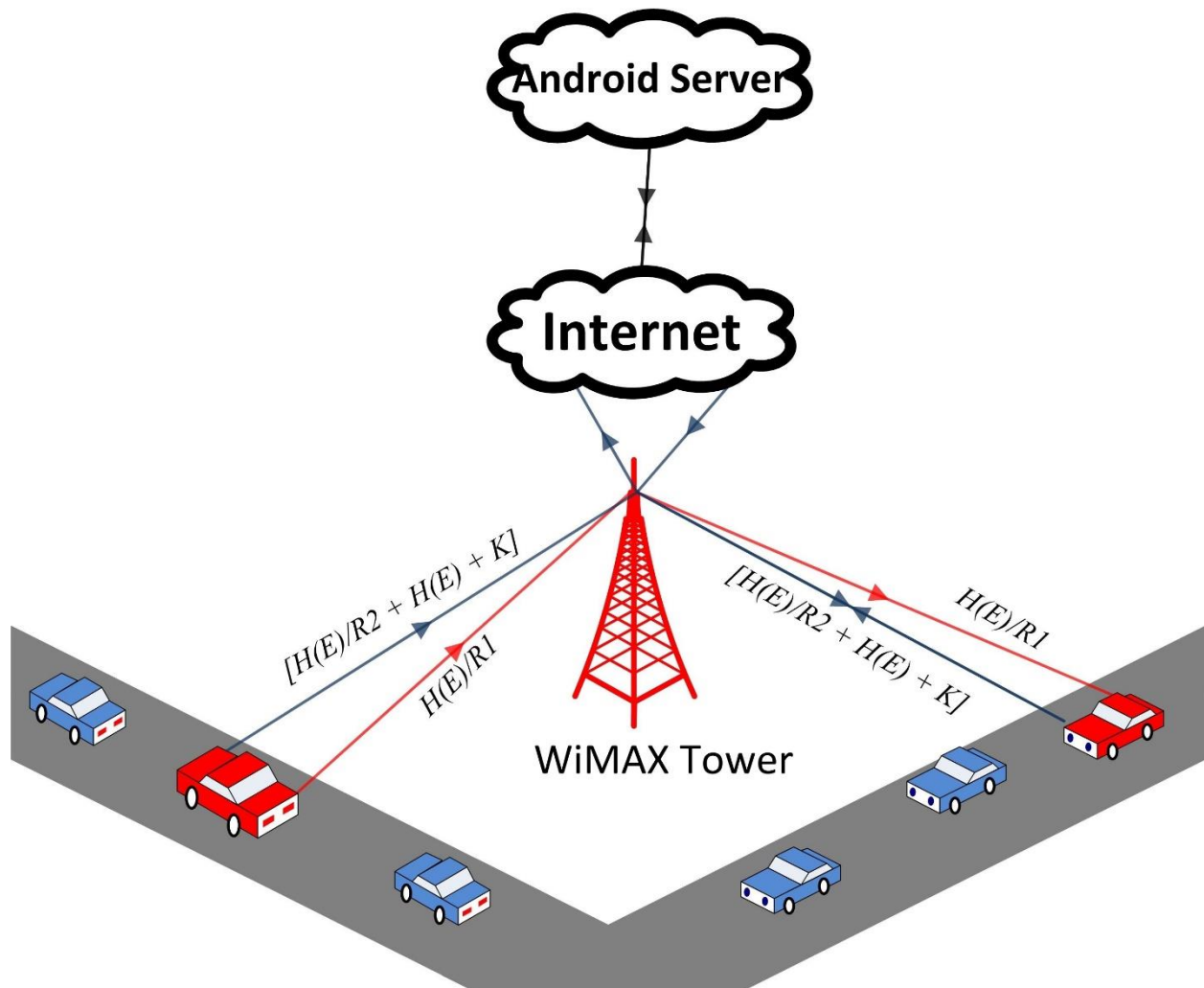


Figure 7: Secure multimedia messaging service in vehicular ad-hoc network (SMMV) communication between vehicles

The communication between the vehicles is an end-to-end encryption where only the end user can decrypt the message. If the encrypted message is compromised before reaching the end user, the user will receive a notification showing that the message is corrupted. Therefore, if the hacker manages to hack the messages he can only see the encrypted part of it, which will protect the confidentiality of the message.

## CHAPTER IV

### APPLICATION DEVELOPMENT

Application development includes six modules.

1. User interface manager
2. Networking manager
3. Server manager
4. Messaging manager
5. Cryptic manager
6. Data manager

#### 4.1 USER INTERFACE MANAGER

The user interacts with the application using the user interface. Once the user is registered to the Android application using the Gmail account, it filters out the contact list and displays the users that have already registered. This module is also responsible for converting the voice messages to text, progress bars, received message notifications and other activities. Speech recognition capabilities were developed using Java speech API.

#### 4.2 DATA MANAGER

The data manager acts as a central system that communicates with other modules. It serves as a primary interface between user interface (UI) and the rest of the modules. It receives the information from the UI manager and communicates with either the networking manager or the cryptic manager depending on the request. Also, it reflects the messages received from the messaging manager, networking manager, and cryptic manager to the UI manager.



### **4.3 MESSAGING MANAGER**

The messaging manager interacts with the data manager (DM) to receive and send the encrypted messages. It always listens for the incoming messages and sends them to DM for further processing. It sends and receives the messages using WiMAX IEEE 802.16. If the network is unavailable, it immediately checks for the messages and sends the unsent messages once it connects back to the network.

### **4.4 CRYPTIC MANAGER**

This module deals with the encryption and decryption aspect of the message. This module receives the message information from the user interface through the data manager. A random key is generated, and the message is encrypted with the AES algorithm using the key and subsequently is converted to SHA-256 digested message. The encrypted message and timestamp are sent to the receiver using messaging manager, while the hashed message is saved in the server database. From the receiver perspective, this module receives the information from messaging manager through data manager. The AES message is converted to the SHA-256 message and compares this hash message with the one stored in the database. If both of them match, then the original message is decrypted by the recipient or else it shows message is corrupted.

### **4.5 SERVER MANAGER**

The server manager handles the backend part of the application. This module interacts with the networking manager and database. Information received from the networking module is authenticated and saved in the database based on the requirement protocols. For example, when a user is registered to the secure multimedia messaging application, the server manager collects this information from the networking manager and creates a schema in the database. Then it

accepts the request and responds accordingly. This module is responsible for database connections, validating and authenticating the user, creating database schemas, and responding to the requests submitted by the user.

#### 4.6 NETWORKING MANAGER

The network manager in this application acts as a bridge between the data manager and the server manager. This module is responsible for the establishment of a connection to the server, user authentication messages, gathering required information from the server to the application (key), updating the user when encrypted messages are sent or received, and other services that UI needs.

Figure 8 explains the interaction between the modules. Each module is dependent on one or more modules.

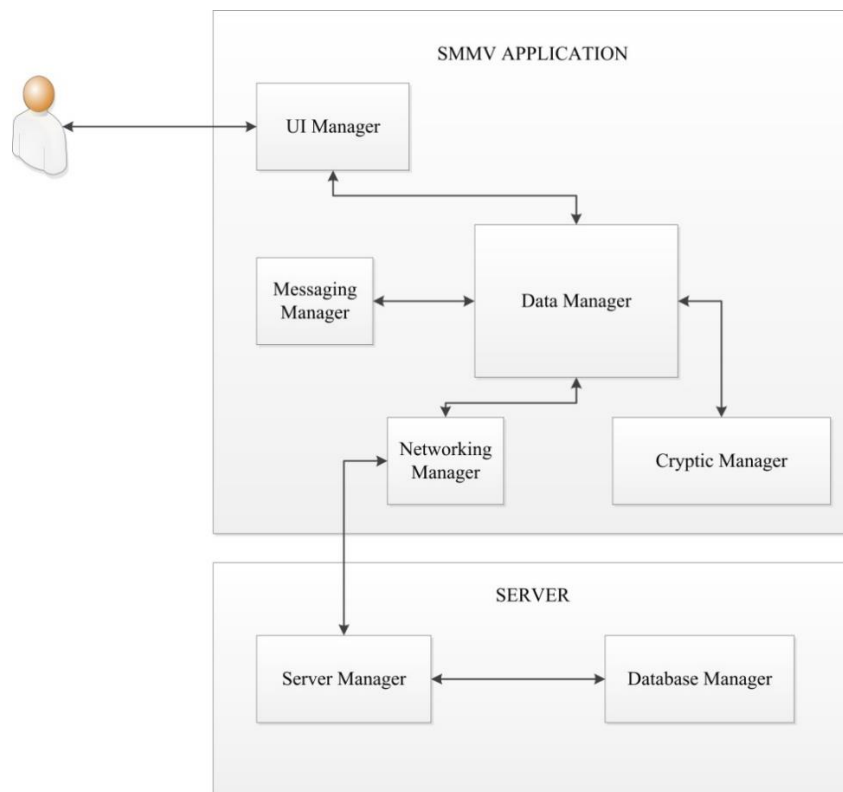


Figure 8: Functional Block diagram showing interactions of modules in the Secure Multimedia Message Application

## 4.7 USING EVENT BUS

The event bus is an Android-based bus that allows publish-subscribe-style communication between modules or components. Instead of waiting for the response from the other component, the information is instead dropped into the bus, which is later picked up by the respective component from the bus. For example, the data manager (DM) wants to communicate with messaging manager (MM), so DM thread has to wait until it gets the response from MM. Data manager cannot start a new thread until MM completes the job. The advantage with the event bus is that, once DM completes the assignment, it can drop into the event bus and start responding to a new request. Additionally, MM can pick the assignment from the event bus, complete it and drop it into the event bus. Later, DM picks it up, processes it and drops into the event bus for the next step.

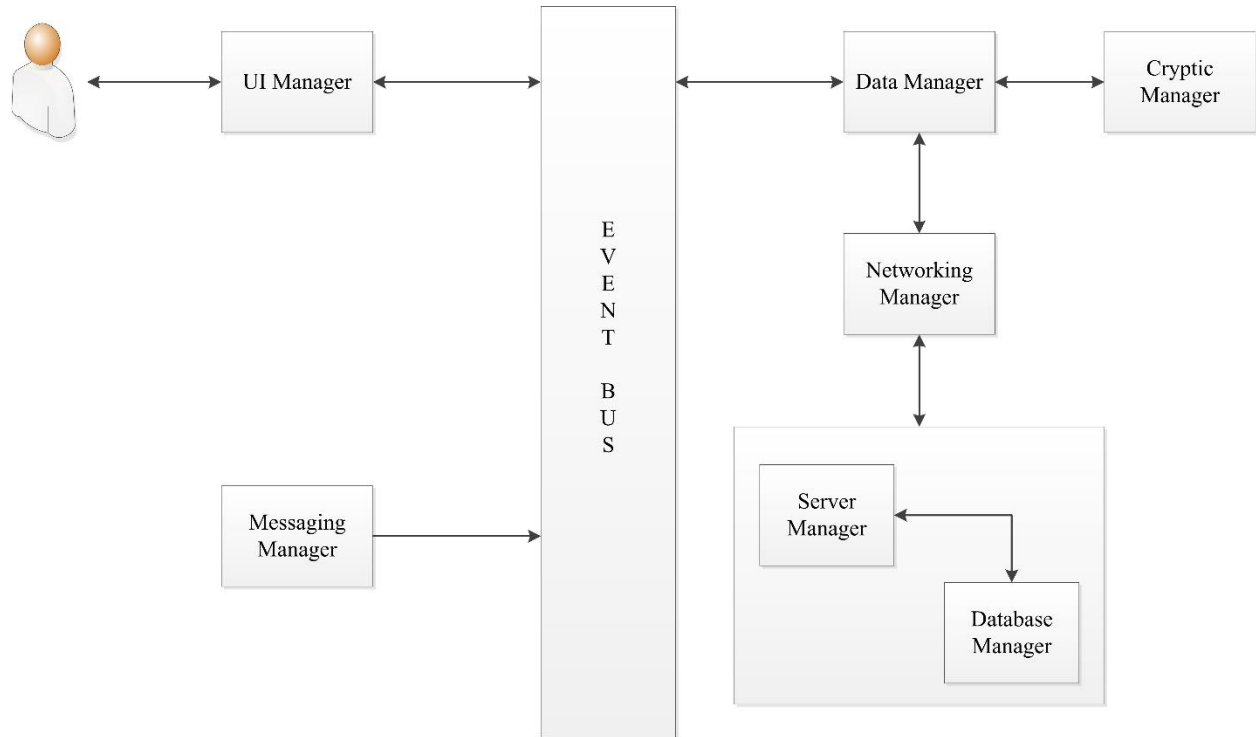


Figure 9: Flow of Events interacting with Event Bus

Figure 9 shows the flow of events from one module to another module. Though it did not eliminate all the dependencies, a few of the modules are independent of each other in order to improve the response time.

#### 4.8 CLASS DIAGRAM

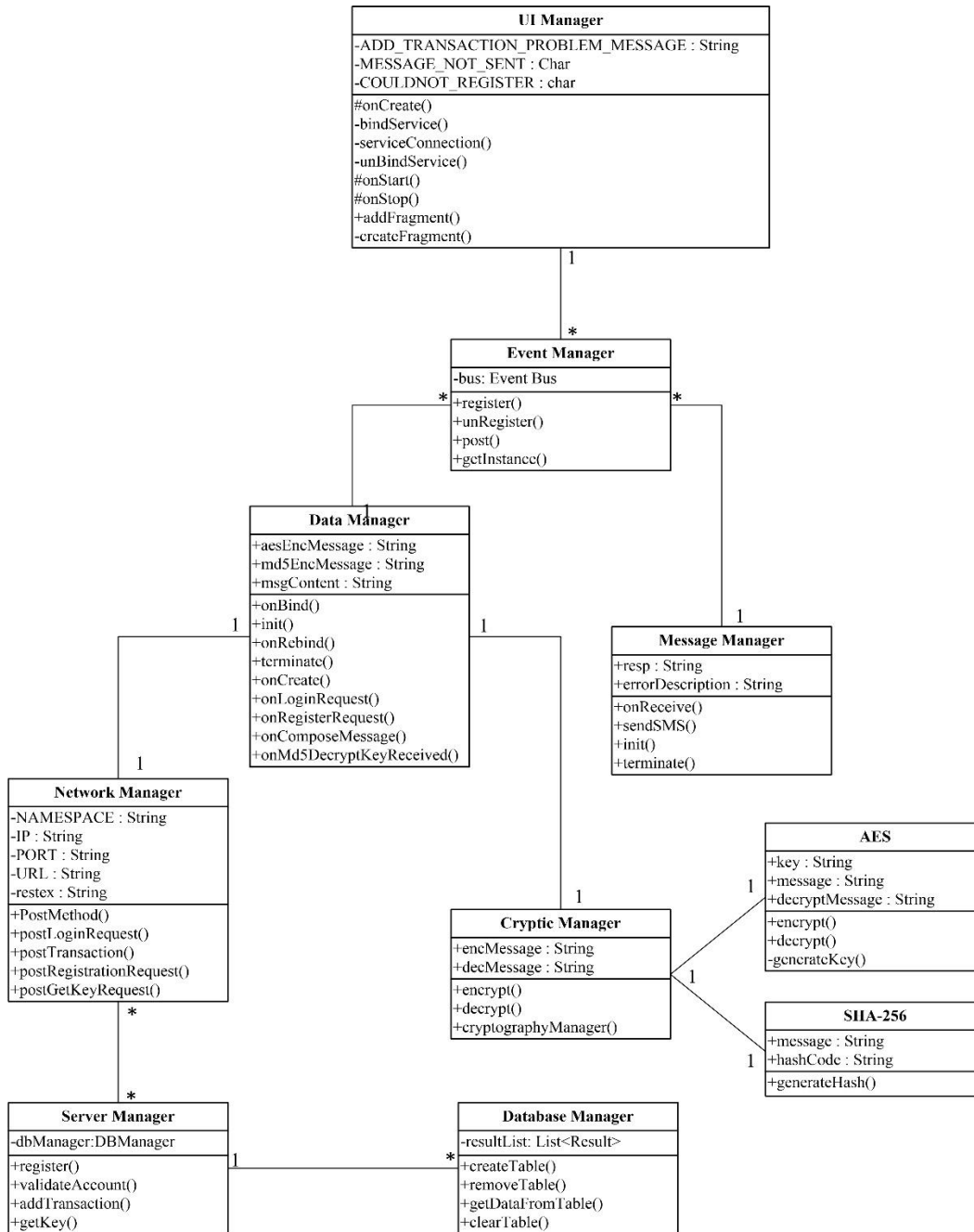


Figure 10: Class Diagram

## Description

The class diagram shows the dependency of each module. The event manager acts as a bus between UI manager, data manager and messaging manager. The data manager manages the flow of events in between other classes shown.

### 4.9 SEQUENCE DIAGRAM

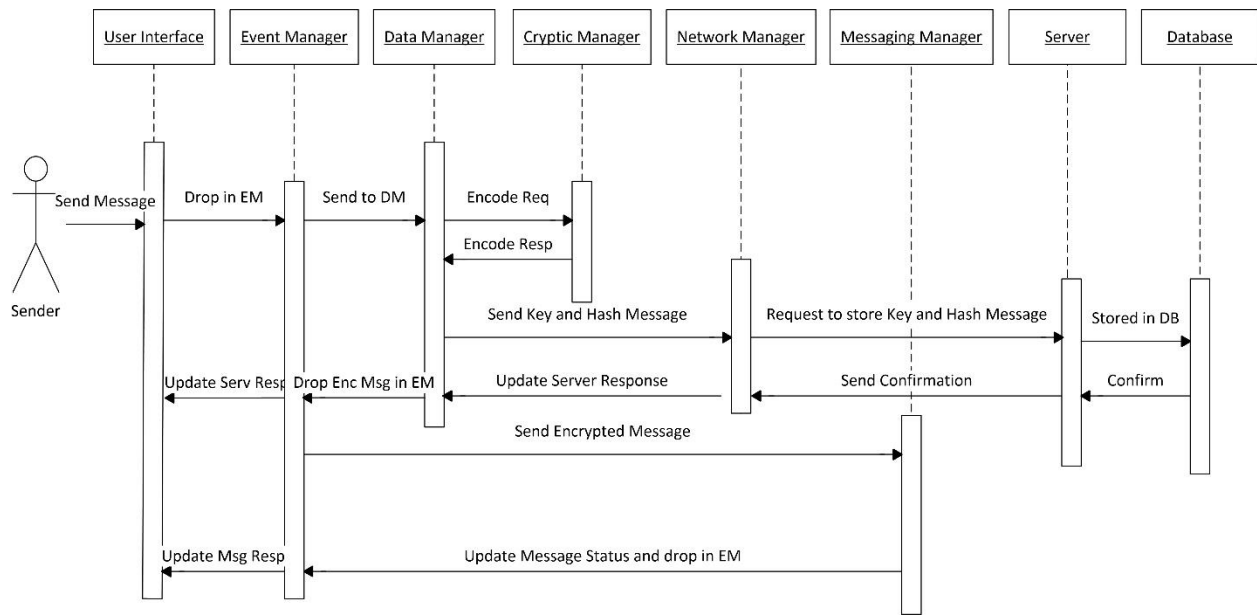


Figure 11: Sequence Diagram showing the events flow when user sends an Encrypted Message

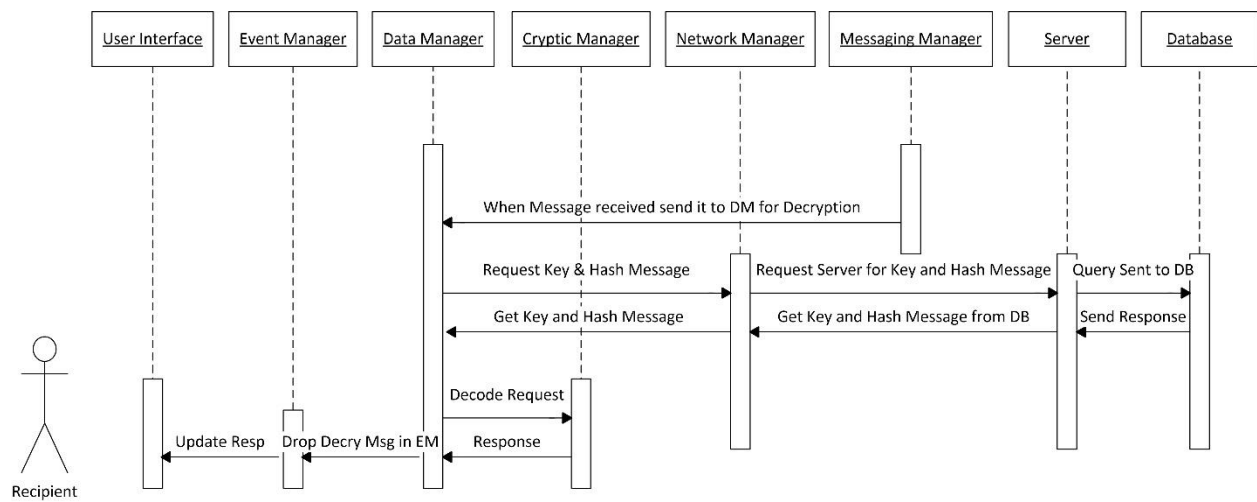


Figure 12: Sequence Diagram showing the events flow when a recipient receives an Encrypted Message

## Description

In the sequence diagram, Figure 11 illustrates the flow of events triggered when a user sends a secure message to a recipient. Figure 12 shows the flow of events when a message is received by the other user.

## CHAPTER V

### SECURITY PERFORMANCE EVALUATION

Only a couple of researchers previously studied secure communications between the vehicles using cryptographic algorithm approaches. Yogesh et al. [34] conducted an analysis using the symmetric key cryptographic system in VANET. They used AES encryption using the same key to encrypt and decrypt. Though the encryption is strong, there are a few challenges that must be surmounted. In their research, they found that, since the VANET network topology is unpredictable, sharing a symmetric key is a challenge; so, a key is exchanged between parties before the start of communication that is not feasible in the current scenario. Also, they mentioned that a key storage overhead was created to authenticate the vehicles from the chain of certificates. Wang et al. [33] proposed a secure communication scheme for VANET for secure communication between vehicles. They used AES encryption for secure message communication. They also used the same approach as before exchanging keys and certificates between the parties before starting the communication. This methodology also exemplified the same disadvantages.

Android application approaches will help in excluding the procedure of exchanging keys and certificates between the vehicles. It works similarly to a basic Android application that is used in a smartphone. Each vehicle has a unique identification number when registering to this application. A unique randomly generated number is given to the vehicle for identification. Every time the application connects to the server it authenticates. Since the authentication is at the server level, there is no need of maintaining the certificates or keys, which can decrease the overhead of the message. Using the AES encryption will increase the security in message communication. The key used for encryption is randomly generated at the time of encryption and

later stored in the server database. Whenever a recipient opens the message, a request is sent to the server to fetch the key. Therefore, exchanging the keys between parties is achieved virtually, and neither of them has to care about the key; the application will take care of it. Once the key is fetched from the database, entry into the server will be deleted.

As discussed earlier, AES encryption is strong enough to encrypt the data, and is impossible for hackers to decrypt without the correct key. Though they cannot decrypt it, if possible, they can modify it. If the listening channel is compromised, both parties can receive scrambled messages that either party cannot understand. Therefore, SHA-256 is used to verify the message integrity. It can digest any size of the message up to 256 bits. After encrypting the message from the sender side, SHA-256 will digest that message and store it in the server database along with the random key. When the recipient receives the encrypted message, it is digested again to create a hashed message and get compared with the original message received along with the key. If the hash messages are similar, then the message will be decrypted, or the user will receive message is corrupted.

Table 2: Showing time taken for encryption and decryption in computational systems

CPU	RAM	Message size (Bytes)	Encryption time ( $\mu$ sec)	Decryption time ( $\mu$ sec)	City Environment time ( $\mu$ sec)	Highway Environment time ( $\mu$ sec)	Variable Speed Environment time ( $\mu$ sec)
2.6Ghz I7 6 <sup>th</sup> gen	16gb	1000	155048	1171	6.55	6.73	6.97
2.6Ghz I7 4 <sup>th</sup> gen	16gb	1000	202648	4032			

Wang et al. [33] evaluated the performance of cryptographic algorithms using a C++, Dot-Net approach. Since Android applications are Java-based, instead of using C++, Dot-Net approach approach, a Java application was built that uses AES and SHA-256 API. The amount of time taken to encrypt and decrypt was recorded and displayed in Table 2.



The total time taken or time complexity for encryption and decryption was measured in microseconds. Thousand bytes' messages and key length (128 bits) were generated randomly during run time. The total time taken for encryption and decryption using 2.6 GHz computer sixth generation I-7 processor and 16 GB ram is approximately 157493  $\mu$ s or 157.493 milliseconds (SMMV [N]). While older generation computers having similar configurations take 197703  $\mu$ s or 197.703 milliseconds (SMMV [O]). From these results, it was clear that the time taken for encryption and decryption is minimal since AES is the best available encryption tool. Thus, it will be a perfect choice for vehicular communications through Android applications.

The time complexity of the encryption algorithm depends on the size of the input. The time taken for encryption and decryption increased with increase in the size of the input data. In Table 3 the encryption and decryption time with variable data sizes was summarized.

Table 3: Showing time taken for encryption and decryption with variable data sizes

In Bytes	Using 2.6 GHz I7 6th gen CPU; SMMV (N)		Using 2.6 GHz I7 4th gen CPU; SMMV (O)	
	Encryption time ( $\mu$ sec)	Decryption time ( $\mu$ sec)	Encryption time ( $\mu$ sec)	Decryption time ( $\mu$ sec)
1000	155048	1171	202648	4032
5000	167108	3838	214210	6050
10000	195364	5177	235579	11507
15000	216151	6506	271209	14322
20000	242256	7697	308342	20651
25000	285474	9105	356668	18926
30000	336986	10602	418563	18230
35000	382935	11649	485581	20771
40000	436229	13506	579594	22655
45000	517647	14553	680974	25693
50000	603843	15833	783266	23001

Figure 13 shows the increase in the total amount of time taken for encryption and decryption with an increase in message size.

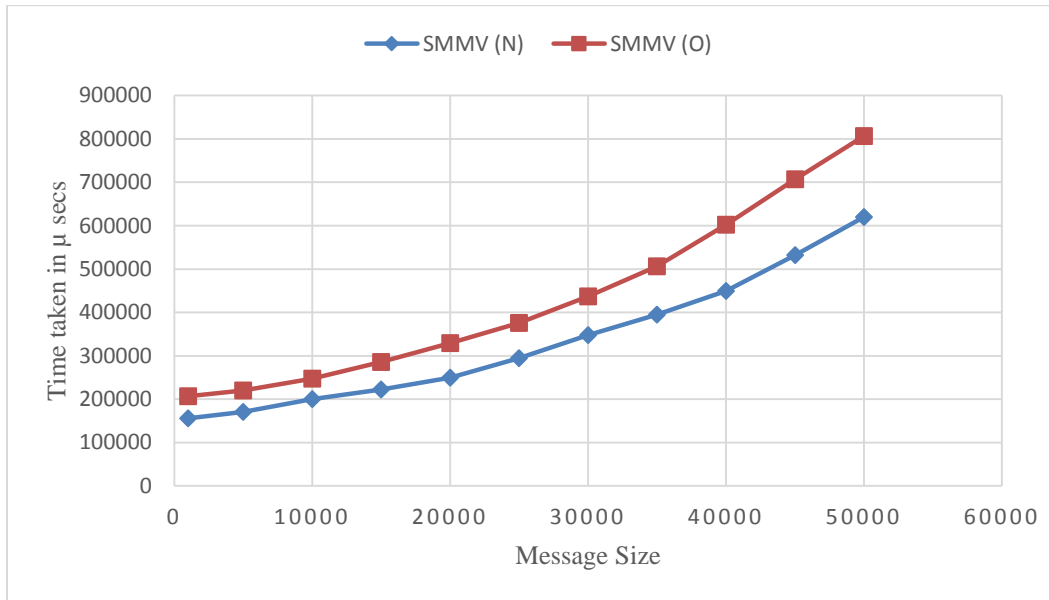


Figure 13: Time taken for Encryption and Decryption with increase in Message size

## CHAPTER VI

### PERFORMANCE ANALYSIS OF MESSAGE TRANSMISSION

Certain authors [22] have analyzed the performance of various routing protocols over WiMAX. As mentioned earlier, they have concluded that AOMDV is the best routing protocol among others in WiMAX-based vehicular message transmission. They have developed a simulation model on NS2 simulator that mimics various vehicular environments including city, highway, and variable speed environment networks. The amount of time taken for message transmission in these environments is 6.55, 6.73, and 6.97  $\mu$ s, respectively. Since interaction with the servers cannot be simulated through the NS2 simulator, the times taken to interact were not included in the results because a variable depends on the network strength and speed.

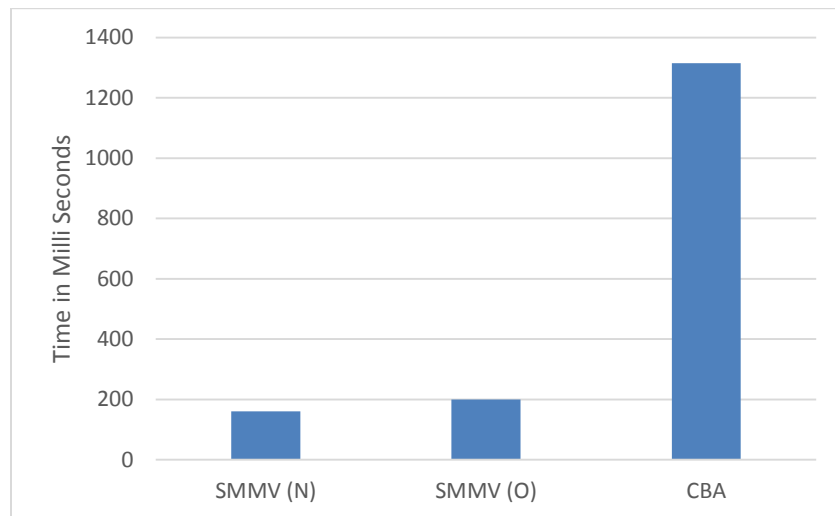


Figure 14: Secure multimedia messaging protocol vs Certificate-based authentication protocol

Certificate-based authentication [34] takes approximately 1.3 secs to verify the authenticity of the message, which does not include message encryption, decryption, and transmission. The total time taken for message encryption, communication, and decryption through the SMMV application takes less than 0.2 secs, which is approximately 85% faster than

CBA. Also, the keys are exchanged virtually so there is no certificate maintenance cost and the application checks the message integrity.

## CHAPTER VII

### PERFORMANCE ANALYSIS OF ROAD SIDE UNIT (RSU) AND WORLDWIDE INTEROPERABILITY FOR MICROWAVE ACCESS (WiMAX)

Only a few researchers analyzed IEEE 802.11p, 802.16e, and 802.16-2004 [36], [37]. Based on their research a simulation model was built to evaluate the performance of RSUs and WiMAX with SMMV protocol. Road side unit uses IEEE 802.11p and WiMAX uses IEEE 802.16 as a standard for effective communication. Road side unit operates at 5.9 GHz frequency with a coverage area ranging 1000 m at 6 Mbps, while WiMAX operates at 2.5 GHz frequency with a coverage area of 50 km at 75 – 300 Mbps. On top of using WiMAX for communication, SMMV protocol uses AES and SHA-256. Ad-hoc on-demand multipath distance vector routing protocol was used in this simulation to find the best route.

In the simulation model, a distinctive urban scenario was selected where roads were 30 m wide and 13 kms long. There was only one WiMAX tower in the simulation, while there were 13 RSUs 1000 m apart spread over the length. Simulation of urban mobility (SUMO) was used to create the data from vehicles. The data were generated and populated into NS2 simulator for pursuant simulations.

#### 7.1 SIMULATION RESULTS

Two environments were selected to run the simulations: city and high-speed highway environments. In the simulation, two cases were evaluated including vehicular speed vs. packet loss ratio and end-to-end delay in message delivery. Fifty vehicles (nodes) were selected in the city environment with a speed ranging from 10 to 50 mph, while 20 vehicles were chosen in highway environments with speed ranging from 50 to 85 mph.

### 7.1.1 Vehicular Speed vs. Packet Loss Ratio

Figures 14 and 15 show the impact of vehicular speed in packet loss in city and highway environments. Results showed that as the speed of the vehicle increased the connective time to RSU decreased and eventually affected the data communication. At low speeds, the percent of packet loss was minimal in RSUs, but, as the speed increased more than 20 mph, the rate of packet loss increased, proportionally.

The upload and download speeds of IEEE 802.11p diminished when compared to WiMAX; this appeared to be the main reason for packet loss. Also, with the rise in vehicular speed, the number of handovers increased that led to greater aggregates of packet loss percentage. While in WiMAX there were no handovers, vehicles were constantly connected to the network which resultantly minimized the loss of data.

Secure multimedia messaging protocol (SMMV) followed a similar pattern of WiMAX. The percent of packet loss in SMMV, when compared to WiMAX, was minimal.

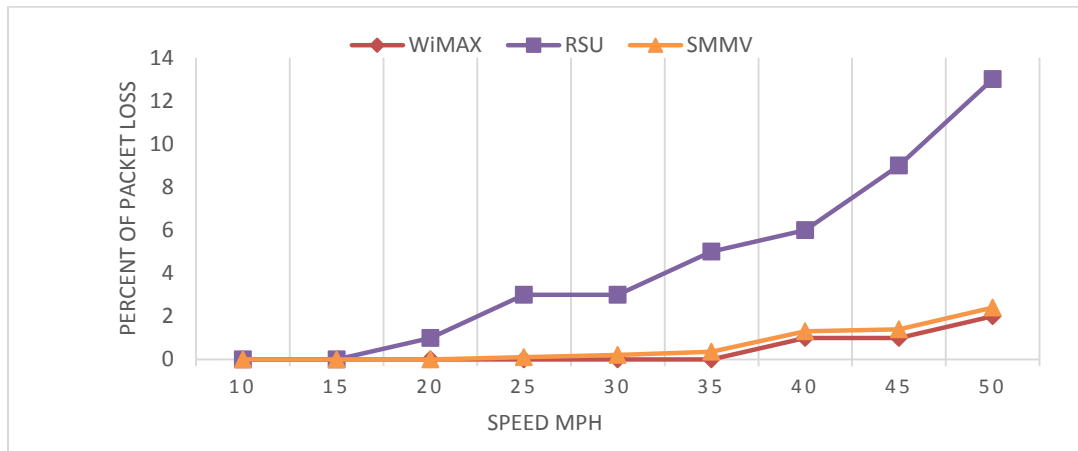


Figure 15: Impact of vehicular speed in percentage of packet loss

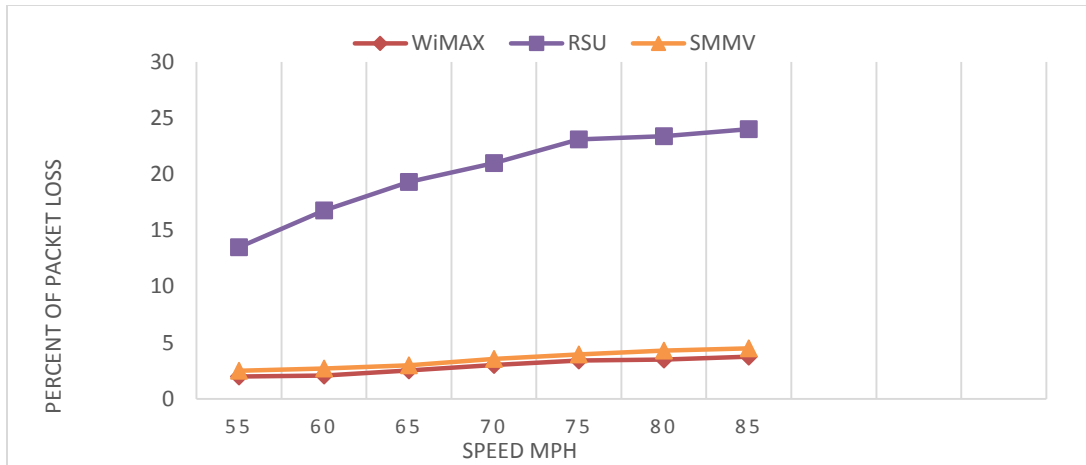


Figure 16: Impact of vehicular speed in percentage of packet loss on highway

### 7.1.2 Vehicular Speed Vs End-to-End Delay

Figures 16 and 17 show the impact of vehicular speed on delays in message delivery in city and highway environments. As the speed increased, delivery delay increased because of handovers in RSU. Conversely, in WiMAX the impact was much less. Compared to WiMAX, the delay in SMMV was in between 0.2 to 0.35 milliseconds.

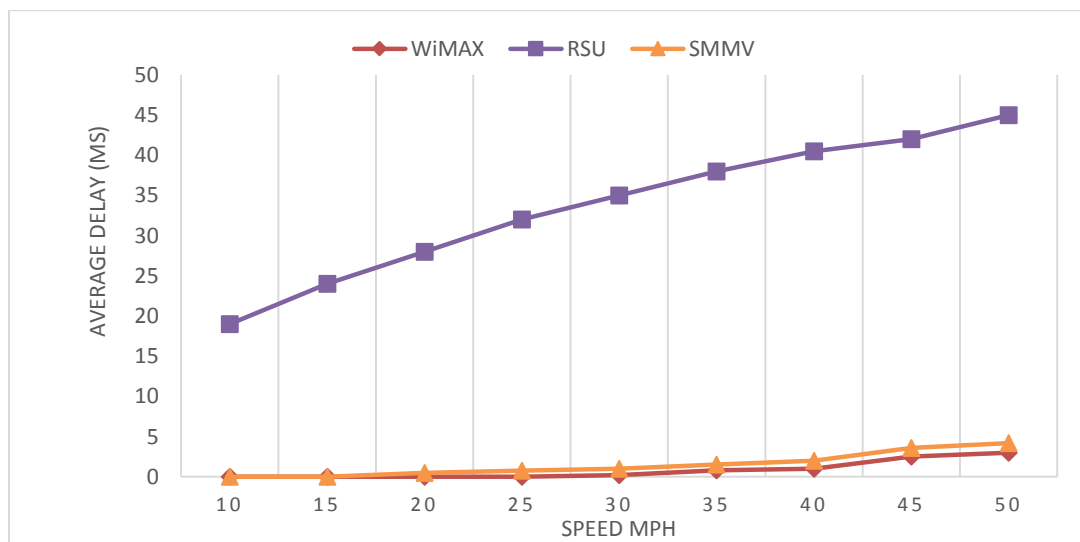


Figure 17: Impact of vehicular speed on message delivery

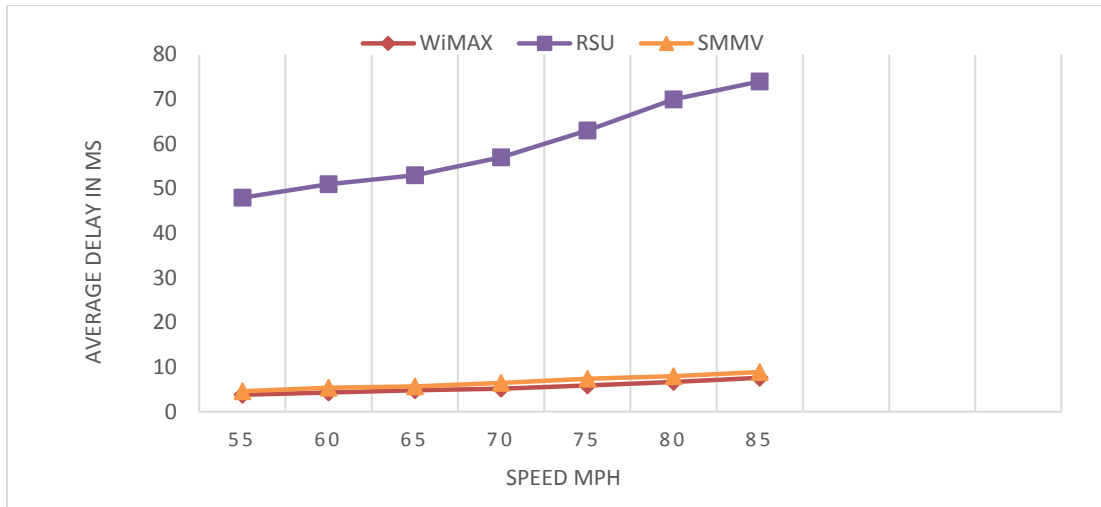


Figure 18: Impact of vehicular speed on message delivery on highway

From the simulation results, it was evident that WiMAX was a better choice than RSUs regarding percent packet loss and average end-to-end delay. Authors of [37] also mentioned that the performance of WiMAX was better than RSU when the number of nodes (vehicles) increased. Also, SMMV protocol followed WiMAX strictly in terms of percent packet loss and message delivery delay.



## CHAPTER VIII

### CONCLUSION

In this thesis, an efficient and secure protocol for communicating personal messages from vehicle to vehicle was proposed. Secure multimedia messaging in vehicular ad-hoc network (SMMV) is an Android application protocol that utilizes WiMAX, rather than RSU, for optimized message communication. From the simulation results, it was evident that WiMAX's percent of packet loss and delay in message delivery is better than RSUs. Advanced encryption standard helps in terms of maintaining the message authenticity and confidentiality while SHA-256 checks for message integrity. These secured protocols provide end-to-end encryption for the users. Compared with the present certificate-based authentication models of message transmission, this protocol was approximately 85% faster with minimal cost for maintenance.

## REFERENCES

- [1] WHO, 27 10 2015. [Online]. Available:  
[http://www.who.int/gho/road\\_safety/mortality/number\\_text/en/](http://www.who.int/gho/road_safety/mortality/number_text/en/).
- [2] National Safety Council, "<http://www.nsc.org/pages/home.aspx>," 06 2015. [Online]. Available: <http://www.nsc.org/NewsDocuments/2015/6-month-fatality-increase.pdf>. [Accessed 28 01 2016].
- [3] Aldana, K., "[www.nhtsa.gov](http://www.nhtsa.gov)," 28 05 2014. [Online]. Available:  
[http://www.nhtsa.gov/About+NHTSA/Press+Releases/2014/NHTSA-study-shows-vehicle-crashes-have-\\$836-billion-impact-on-U.S.-economy,-society](http://www.nhtsa.gov/About+NHTSA/Press+Releases/2014/NHTSA-study-shows-vehicle-crashes-have-$836-billion-impact-on-U.S.-economy,-society). [Accessed 24 03 2016].
- [4] Wenshuang, L., Zhuorong, L., Hongyang, Z., Rongfang, B., and Yunchuan, S., "Vehicular Ad Hoc Networks: Architectures, Research Issues, Methodologies, Challenges, and Trends," *International Journal of Distributed Sensor Networks*, vol. 2015, pp. 1-11, (2015).
- [5] Suganya, R., Lalitha, S., and Kuppusamy, P., "Intelligent transport system communication using security privacy data with WiMAX," *International Journal of Infinite Innovations in Engineering and Technology*, vol. 2, no. 6, pp. 26-30, (2015).
- [6] Kiho, L. and Manivannan, D., "An efficient protocol for authenticated and secure message delivery in vehicular ad hoc networks," *Vehicular Communications*, vol. 4, pp. 30-37, (2016).
- [7] Chris, G. and Shahid, M., "A Survey of VANET Technologies," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 5, no. 9, pp. 1-11, (2014).
- [8] Mojela, L. and Booyesen, M., "On the use of WiMAX and Wi-Fi to provide in-vehicle

- connectivity and media distribution," in *IEEE International Conference on Industrial Technology (ICIT)*, (2013).
- [9] Anna, V., Mauro, B., and Roberto, C., *Smart Vehicles, Technologies and Main Applications in Vehicular Ad hoc Networks*, *Intech Open*, (2013).
- [10] "android.com," Android, [Online]. Available: <https://www.android.com/auto/>. [Accessed 23 10 2015].
- [11] "apple.com," Apple, [Online]. Available: <http://www.apple.com/ios/carplay/>. [Accessed 23 10 2015].
- [12] Google, "Android Compatibility Definition," Google Inc, (2015).
- [13] Xiong, H., Beznosov, K., Qin, Z., and Ripeanu, M., "Efficient and spontaneous privacy-preserving protocol for secure vehicular communication," in *Proceedings of 2010 IEEE International Conference on Communications, ICC*, (2010).
- [14] Huang, L., Jie, L., and Mohsen, G., "A novel id-based authentication framework with adaptive privacy preservation for VANETs," in *Proceedings of Computing, Communications and Applications Conference*, Hong Kong, (2012).
- [15] Xiaodong, L. and Xu, L., "Achieving efficient cooperative message authentication in vehicular ad hoc network," *IEEE Transactions on Vehicular Technology*, pp. 3339-3348, (2013).
- [16] Yong, H., Tingting, H., and Yu, C., "A cooperative message authentication protocol in VANETs", in *Proceedings of Global Communications Conference*, Anaheim, (2012).
- [17] Shuang. Y, "www.standards.ieee.org," IEEE, [Online]. Available: <http://standards.ieee.org/news/2011/80216m.html>. [Accessed 17 02 2016].

- [18] Vinoth, V. and Monica, C., "A sama scheme for improving qos in 4g multihops wireless networks", *ARNP Journal of Engineering and Applied Sciences*, vol. 10, no. 7, pp. 2907-2913, (2015).
- [19] Johnson, D. and Maltz, D., "Dynamic Source Routing in Ad Hoc Wireless Networks," *The Kluwer International Series in Engineering and Computer Science*, vol. 353, pp. 153-181, (1996).
- [20] Charles, P. and Elizabeth, R., "Ad-hoc on-demand distance vector routing," in *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, (1999).
- [21] Marina, M. and Das, S., "Ad hoc On-Demand Multipath Distance Vector Routing," *Wireless communications and mobile computing*, vol. 6, no. 7, pp. 969-988, (2006).
- [22] Dorge, P. and Dorle, S., "Performance Analysis of WiMAX Based Vehicular Ad hoc Networks with Realistic Mobility Patterns," *International Journal of Computer and Communication System Engineering*, vol. 2, no. 5, pp. 641-653, (2015).
- [23] Android, "https://source.android.com," Google, [Online]. Available: <https://source.android.com/security/#android-platform-security-architecture>. [Accessed 16 03 2016].
- [24] Android, "www.android.com," Google, [Online]. Available: <https://developer.android.com>. [Accessed 16 03 2016].
- [25] Zou, Y., Zhu, J., Wang, X., and Hanzo, L., "A Survey on Wireless Security: Technical Challenges, Recent Advances and Future Trends," *Accepted to Appear in Proceedings of the IEEE*, (2016).
- [26] NIST, "www.nist.org," NIST, 21 01 2006. [Online]. Available:

[http://www.nist.org/nist\\_plugins/content/content.php?content.39](http://www.nist.org/nist_plugins/content/content.php?content.39). [Accessed 16 01 2016].

- [27] Nechvatal, J., Barker, E., Bassham, L., Burr, W., Dworkin, M., and Roback, J., "Report on the Development of the Advanced Encryption Standard (AES)," *National Institute of Standards and Technology*, (2000).
- [28] Kak, A., "Lecture 8: AES: The Advanced Encryption Standard," 12 04 2016. [Online]. Available: <https://engineering.purdue.edu/kak/.../NewLectures/Lecture8.pd>. [Accessed 21 06 2016].
- [29] Florent, C. and Antoine, J., "Differential collisions in SHA-0," *Advances in Cryptology*, vol. 1462, pp. 56-71, (2006).
- [30] Xiaoyun, W., Hongbo, Y., and Yiqun, Y., "Efficient Collision Search Attacks on SHA-0," *Advances in Cryptology*, vol. 3621, pp. 1-16, (2005).
- [31] Xiaoyun, W., Hongbo, Y., and Yiqun Y., "Finding Collisions in the Full SHA-1," *Advances in Cryptology*, vol. 3621, pp. 17-36, (2005).
- [32] Criptografia, "Research Gate," [Online]. Available: <https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=8&cad=rja&uact=8&ved=0ahUKEwijnM3o6YrPAhWe8oMKHaA5Ao4QFghSMAC&url=https%3A%2F%2Fwww.researchgate.net%2Ffile.PostFileLoader.html%3Fid%3D534b393ad3df3e04508b45ad%26assetKey%3DAS%253A2735148446228>. [Accessed 12 05 2016].
- [33] Neng-Wen, W., Yueh-Min, H., and Wei-Ming, C., "A novel secure communication scheme in vehicular ad hoc networks," *Computer Communications*, vol. 31, no. 12, pp. 2827-2837, (2008).
- [34] Yogesh, S., Avichal, K., and Manish C. "Analysis of Symmetric Key Cryptosystem in

VANET," *Int. J. on Recent Trends in Engineering and Technology*, vol. 7, no. 2, pp. 63-67, (2012).

- [35] Saira, G., Farrukh, S., Amir, Q., and Rashid, M., "A Survey on Security in Vehicular Ad hoc Networks," *Springer Berlin Heidelberg*, (2013).
- [36] Raúl, A., Antonio, G., and Arthur, E., Comparative Analysis of IEEE 802.11p and IEEE 802.16-2004 Technologies in a Vehicular Scenario, InTech, (2011).
- [37] Bhakthavathsalam, R., Starakjeet, N., "Operational Inferences on VANETs in 802.16e and 802.11p with Improved Performance by Congestion Alert," in *Consumer Communications and Networking Conference (CCNC)*, 2011.
- [38] Nover, H., "Algebraic cryptanalysis of aes: an overview," in *University of Wisconsin*, Wisconsin, 2005.

## VITA

Satya Sridhar Karanki was born in Kakinada, Andhra Pradesh, India, on 10th November 1984, the first son of Venkateswara Rao and Vijaya Lakshmi. After completing 12th grade at Guntur Vikas Junior College, Hyderabad, Telangana, India, he entered into Nagarjuna Univeristy to pursue his Bachelor of Technology in Biotechnology. In May 2006, he successfully completed it. In spring 2007, he started pursuing his Master of Science in Biotechnology at the University of Texas at San Antonio and completed in fall 2009. Later in 2010 he joined as a research associate at Texas A&M Health Science Center, Irma Lerma Rangel College of Pharmacy, Kingsville. Due to his passion toward programming languages, in spring 2013 he started pursuing Master of Science in Computer Science at Texas A&M University-Kingsville.