UNIVERSITY OF CALIFORNIA,
IRVINE

Physical Layer Key Generation for Wireless Communication Security in Automotive Cyber-Physical Systems

THESIS

submitted in partial satisfaction of the requirements
for the degree of

MASTER OF SCIENCE

in Computer Engineering

by

Anthony Bahadir Lopez

Thesis Committee:
Professor Mohammad Al Faruque, Chair
Professor Fadi Kurdahi
Professor Ozdal Boyraz

2017

DEDICATION

To those who believe in me.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I am grateful for the feedback, support, and mentorship of Professor Mohammad Al Faruque, Professor Fadi Kurdahi, and Professor Ozdal Boyraz.

I would like to thank Kimmy and those who supported and believed in me: friends, family members, lab mates, and mentors.

I would also like to thank the National Science Foundation for their financial assistance in my graduate studies through the NSF Graduate Research Fellowship Program and the Bridge to Doctorate Program.

Finally, I would like to thank ACM, IEEE, and the Optical Society of America for permitting me to use their copyrights and allowing me to publish in their conferences and journals.

The text of this thesis is a reprint of the material as it appears in “Exploiting Wireless Channel Randomness to Generate Keys for Automotive Cyber-Physical System Security”, in Proceedings of the 7th International Conference on Cyber-Physical Systems (ICCPS ’16) [53], “A Physical Layer Security Key Generation Technique for Inter-Vehicular Visible Light Communication”, in Advanced Photonics 2017 [59] and “Physical Layer Key Generation: Securing Wireless Communication in Automotive Cyber-Physical Systems”, accepted to be published in Transactions on Cyber-Physical Systems 2018 (TCPS ’18) [54]. The co-authors listed in these publications directed and supervised research which forms the basis for the thesis.

ABSTRACT OF THE THESIS

Physical Layer Key Generation for Wireless Communication Security
in Automotive Cyber-Physical Systems

By

Anthony Bahadir Lopez

Master of Science in Computer Engineering

University of California, Irvine, 2017

Professor Mohammad Al Faruque, Chair

Modern automotive Cyber-Physical Systems (CPS) are increasingly adopting a variety of wireless communications (Radio Frequency and Visible Light) as a promising solution for challenges such as the wire harnessing problem, collision detection and avoidance, traffic control, and environmental hazards. Regrettably, this new trend results in security challenges that can put the safety and privacy of the automotive CPS and passengers at great risk. Further, automotive wireless communication security is constrained by strict energy and performance limitations of electronic controller units and sensors. As a result, the key generation and management for secure automotive wireless communication is an open research challenge. This thesis aims to help solve these security challenges with a novel key management scheme built upon a physical layer key generation technique that exploits the reciprocity and high spatial and temporal variation properties of the automotive wireless communication channel. A key length optimization algorithm is also developed to help improve performance (in terms of time and energy) for safety-related applications. Channel models, simulations and real-world experiments with vehicles and remote-controlled cars were performed to validate the practicality and effectiveness of the scheme. Lastly, it is shown that generated keys may have high security strength (67% min-entropy for the Radio Frequency domain and high randomness according to NIST tests for the Visible Light domain) and that code size overhead is 20 times less than state-of-the-art security techniques.

Chapter 1

Introduction

1.1 Background

Wireless technologies are widely implemented in automotive Cyber-Physical Systems (CPS) for navigation schemes (e.g., GPS) and infotainment applications such as hands-free calling, and satellite radio [14]. As a light-weight solution to the wireless harnessing problem [28] and for its aforementioned applications, wireless technologies applied on many microcontrollers all throughout a vehicle can enable a powerful improvement in safety and comfort for people and functionality and efficiency for automotive CPS [13, 15]. A notable example is the Tire Pressure Monitoring System (TPMS), which is implemented in many modern vehicles and utilizes several controllers and tire sensors to measure and display tire temperatures and pressures to passengers. Through the TPMS, passengers can understand from warning signals when to re-inflate or replace their tires, leading to evasion of unnecessary dangers.

An astounding 80% of all vehicular[1] collisions are caused by drivers but it is clear that

[1] The contents of our proposed techniques can apply to all types of automotive cyber-physical systems, however this thesis focuses on the average vehicle (e.g., car, truck, or any other thing used to transport people or goods) as a motivating example.

wireless technologies can greatly reduce the risk of driver-caused collisions and improve traffic efficiency [56]. In order to realize these objectives, federal agencies (e.g., United States Department of Transportation) and research organizations (e.g., Google) are developing general wireless vehicular communication protocols (V2X) in Radio Frequency (RF) and Visible Light (VL) mediums. Although RF is the more developed and invested wireless communication medium, Visible Light Communication (VLC) is a rapidly growing wireless optical communication technology which exploits the advantage of omnipresent LEDs and photodiodes. For short-range communication, VLC can be an effective alternative and companion to RF-based wireless communication due to its high spectral availability, precise directional Line of Sight (LOS) propagation, and immunity to multipath fading. Yet, VLC also faces security threats including jamming, eavesdropping, interception and physical infrastructure attacks [6]. Whichever medium is used, protocols for V2X are categorized under Intra-Vehicular, Vehicle-to-Vehicle (V2V) or Vehicle-to-Infrastructure (V2I) communication protocols [19]. With V2X communication, vehicles can perform collision detection and prevention, path control, traffic management, environmental hazard avoidance, and new forms of entertainment through Internet connection. Below, Figure 1.1 provides an illustration of a setting with V2X communication.

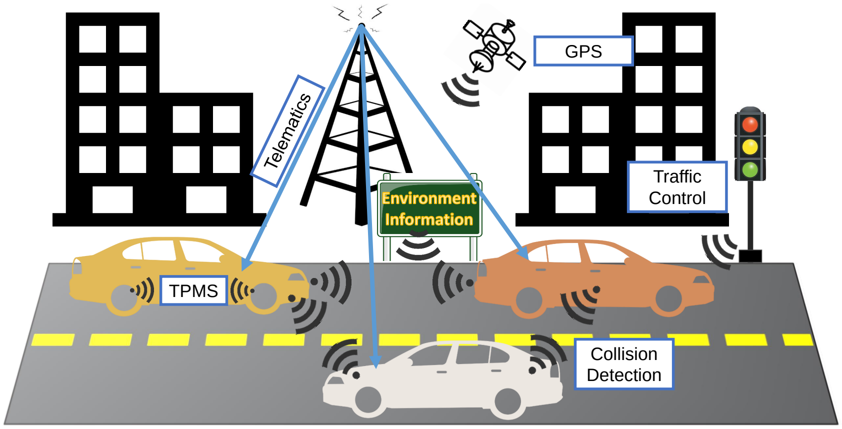


Figure 1.1: Examples of V2V and V2I Applications

1.2 Motivation

With a new model of interconnectivity, both old and new V2X applications will connect traditionally isolated vehicles to each other, infrastructure, satellites and other entities through insecure wireless channels. Already, security concerns have arisen over leakage of critical information about the vehicle or the passengers and over the possibility of indirect control of the vehicle's mechanisms [57, 25, 10, 33]. In fact, these security concerns date back to the mid-1990s, a time where many vehicles used Remote Keyless Entry (RKE). Researchers eventually found that RKE was vulnerable to relay attacks, where relaying a signal through simple technology could unlock and start a vehicle when its owner is away [9]. In 2005, a Texas Instrument RFID transponder implemented in millions of vehicles was found to be hackable, thus portraying another security threat [9]. Then in 2010, researchers developed an attack that captured and read TPMS communication packets from a vehicle up to 40 meters (.02 miles) away. Furthermore, they demonstrated the possibility of injecting packets into the TPMS network to trigger a fake warning signal [22]. In the recent years of 2014 and 2015, researchers with support from the Defense Advanced Research Projects Agency (DARPA), have developed and demonstrated exploitable hacks regarding vehicular infotainment applications and systems like UConnect [33]. Their hacking demonstrations ended in the recall of many vulnerable vehicles, such as Chrysler [34]. An introduction of V2X communication will eventually cause similar and new security concerns. As a result, researchers such as those from the European Telecommunications Standards Institute (ETSI) are proposing the following security objectives for V2X communication: confidentiality, integrity, availability, accountability and authenticity (for more details, please see the technical reports [16]).

This thesis focuses on the security requirement of confidentiality. We summarize that for wireless communication in automotive CPS, messages will need to be encrypted depending on the confidentiality requirements of applications [46]. As a simple example, account information will need to be encrypted in financial applications like Electronic Toll Collection

(ETC) [42, 25] and for cooperative pre-crash sensing [12]. Another major challenge for V2X communication is authentication. Since users are exposed to many dangers due to the wireless communication, it is necessary for a receiver to verify that a transmitted message was generated by a legitimate user. Recently, researchers have proposed to solve the authentication problem in automotive communication in an Ad-Hoc manner [27, 11]. However, this type of scheme requires an established secure channel for exchanging authentication information such as secret keys and identifications before communication. It is important to note that these and other security objectives apply to resource-limited (in terms of computational power, energy consumption and memory size) time-critical embedded devices (e.g., sensors, V2X) and resource-limited non-time-critical devices (e.g., infotainment) within the vehicles. Because of their important role in keeping passengers and drivers safe, resource-limited and time-critical devices are the focus of this thesis.

1.3 Overview of Contributions

To address the challenges, a novel physical layer key management scheme is proposed to help secure automotive wireless communication. The scheme is generic and can be applied in different vehicular wireless communication protocols that are Radio Frequency-based or Visible Light-based. The scheme is based upon a physical layer technique that generates symmetric cryptographic keys from the physical randomness of the wireless channel. This technique is suitable for the vehicular wireless communication domain due its lenient requirements on time, memory, and processing power. Key generated from the technique are called Pre-Shared Keys (PSKs) and they can be used as or extended into longer keys that will be used for encryption or authentication (which requires random numbers) purposes. A PSK is a known as a shared secret and is a random bit string known only to a pair of communicating parties.

The biggest advantage of this technique is that it solves the key generation and exchange problem at the same time, which means it can generate the PSKs with high entropy while eliminating the costly requirements of the asymmetric algorithms for the key exchange process. Moreover, it may also replace the asymmetric algorithm for exchanging secret keys and identification for authentication purposes.

This thesis aims to demonstrate, through realistic automotive modeling, simulation and experiments, that higher levels of entropy and performance may be obtained from the moving and changing environment to practically generate symmetric secret keys for automotive CPS wireless communication. The contributions of this thesis are as follows:

1. **A literature survey of state-of-the-art security techniques and related works (Chapter 2)**.
2. **Automotive wireless communication system models for both Radio Frequency-based and Visible Light-based protocols (Chapter 3)** which includes:
   (a) **RF-based wireless channel and attack models from a security perspective (Section 3.1)**.
   (b) **VL-based wireless channel and attack models from a security perspective (Section 3.2)**.
3. **A physical layer key management scheme for automotive wireless communication (Chapter 4)** which includes:
   (a) **A wireless channel-based PSK generation technique (Section 4.1)**.
   (b) **A PSK length optimization technique (Section 4.2)** under constraints based on the scenario of the vehicular communication session.

   (c) **Cryptographic key derivation method (Section 4.3)** which converts the PSK into a suitable key for encryption or authentication purposes.

4. **Simulations and real world experiments (Chapter 5)** to validate the effectiveness and practicality of the models and proposed scheme.

Chapter 2

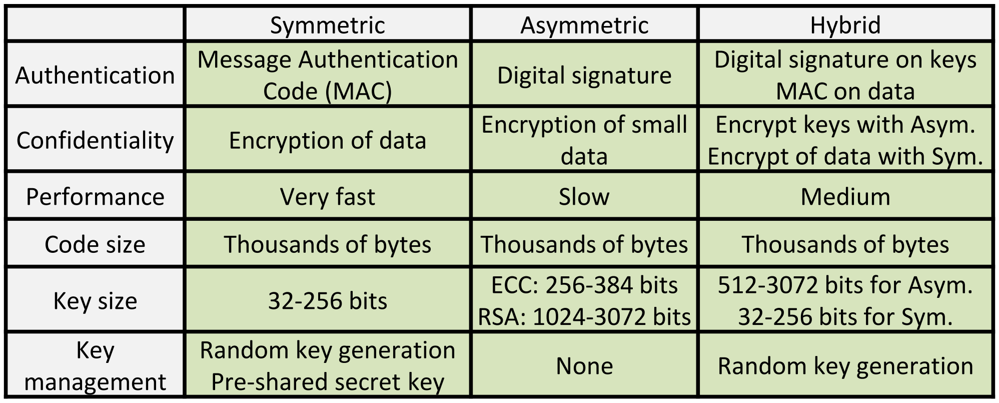
Literature Survey

2.1 State-of-the-Art Key Management Schemes

A typical automotive design needs to provide security for about 20 years or more [46, 52], implying the necessity of a reliable and efficient cryptographic scheme to achieve some of the aforementioned security objectives. Cryptographic algorithms fall under two categories: **1)** Symmetric and **2)** Asymmetric. As seen in Table 2.1, symmetric algorithms, such as the Advanced Encryption Standard (AES), have very high performance and lower energy overhead [39] in comparison to asymmetric algorithms, such as RSA and Elliptic Curve Cryptography (ECC). However, both of these schemes are challenging to implement on resource-limited and time-critical devices.

The major problem of using a symmetric encryption algorithm is that both parties must have a shared secret key to establish a secure communication. On the other hand, although asymmetric algorithms do not require a shared secret key for secure communication, they are too slow for the majority of time-critical applications and too resource-intensive in terms of computational power and memory usage [46, 39, 36]. As a result, higher performance processors have been used to address these issues. However, using such processors (e.g., Qualcomm Snapdragon 602A for V2X) comes with a non-negligible cost (for example, the Qualcomm Snapdragon 602A may involve around $1000 or more in extra cost). Moreover, there are up to 100 Electronic Control Units (ECUs) in a modern car, and many of these ECUs are low cost micro-processors. An alternative security method would be necessary to enable V2X applications on these processors. In some of the state-of-the-art approaches, research groups and government organizations have proposed the use of hybrid solutions to reduce overhead from the asymmetric algorithms [46, 47]. In a hybrid solution, a symmetric key is generated from a Pseudo Random Number Generator (PRNG) or a Key Encapsulation Mechanism (KEM) [20] and exchanged through an asymmetric algorithm. Afterward, higher performance can be achieved through symmetric encryption of data.

Table 2.1: Comparison of Existing Cryptographic Algorithms



	Symmetric	Asymmetric	Hybrid
Authentication	Message Authentication Code (MAC)	Digital signature	Digital signature on keys MAC on data
Confidentiality	Encryption of data	Encryption of small data	Encrypt keys with Asym. Encrypt of data with Sym.
Performance	Very fast	Slow	Medium
Code size	Thousands of bytes	Thousands of bytes	Thousands of bytes
Key size	32-256 bits	ECC: 256-384 bits RSA: 1024-3072 bits	512-3072 bits for Asym. 32-256 bits for Sym.
Key management	Random key generation Pre-shared secret key	None	Random key generation

However, there are still three major limitations to the current hybrid approach: **1)** It requires a key exchange session which uses an asymmetric algorithm whose lengthy computation time is generally not acceptable for safety-related applications which require a reaction time of 50 to 200 milliseconds [46]. **2)** The hybrid solution requires an implementation of the asymmetric algorithm in the embedded devices, thus causing non-negligible memory space overhead. **3)** Similar to symmetric algorithms, the hybrid solution generally relies on a Pseudo Random Number Generator (PRNG) or user-given inputs to help produce a symmetric key with high entropy. These approaches, however, cannot provide enough entropy[1] due to high levels of predictability of the seed or user-given inputs and deterministic nature of the key generation algorithm [37]. **For the aforementioned reasons, secret key generation and exchange are considered challenging problems for automotive wireless applications.**

2.2 Key Generation from Physical Randomness

To help solve this problem of developing a reliable yet efficient and fast encryption mechanism, researchers have been looking toward physical randomness as a high entropy source. As an example, researchers proposed the use of physical randomness (e.g., timing delays, memory values) in circuit characteristics to generate secret keys [44, 50]. Similarly, it is possible to exploit the physical randomness from wireless communication channel characteristics, such as the multipath-induced fading and shadow fading to generate strong secret keys. Most of the state-of-the-art theories and practical methods for generating secret keys using physical characteristics of the wireless channel (or the physical layer) have been proposed within just the last decade but have not been applied to the automotive CPS environment [7, 31, 58, 61, 55, 43, 23, 38, 60].

The success of generating secret keys based on the wireless channel's physical randomness depends on three properties: **1)** Reciprocity of the wave propagation, **2)** Temporal variation, and **3)** Spatial variation in the environment. Besides most of the theoretical works [7, 31], some practical implementations for sensor network applications [43, 23, 1] have been performed and rely on the Multiple-Input and Multiple-Output (MIMO) approach or collaborations among multiple wireless nodes to create secret keys with high entropy. Work in [60] has also provided an implementation for V2X applications. However, it mainly focuses on

[1] entropy (more specifically Shannon Entropy) can be used as a quantified value of randomness for a set of bits.

comparing their algorithm with other key generation algorithms and modeling the spatial and temporal variations of the automotive wireless channel. Moreover, the authors do not consider practical challenges such as abiding by real-time requirements for safety-critical V2V applications and optimizing their algorithm parameters in terms of resources such as time, energy, and memory. Lastly, other types of physical layer security have been discussed in [36] but have not yet been applied to the automotive CPS domain to enable efficient and reliable security protocols as this work has done.

In summary, solving the limitations of the above-mentioned state-of-the-art approaches to secure wireless communication in automotive CPS poses the following challenges:

1. **Finding a reliable high entropy source** to generate secret keys for symmetric cryptographic algorithms.
2. **Designing a reliable solution** for the management of symmetric secret keys.
3. **Optimization of the solution and key size** in terms of performance.

Chapter 3

System Modeling

3.1 Vehicular Radio Frequency Wireless Communication

3.1.1 System Model

Below is a sender-to-receiver model of an automotive wireless communication system, where an ECU or sensor-node inside a vehicle A is communicating with a device from another vehicle or infrastructure B in the presence of an eavesdropper in vehicle E. In this model, the sending signal S_A from A over the wireless channel will be received by B and E as follows:

$$\begin{aligned} R_{A\rightarrow B}(t) &= H_{A\rightarrow B}(t) \times S_A(t) + N_{A\rightarrow B}(t); \\ R_{A\rightarrow E}(t) &= H_{A\rightarrow E}(t) \times S_A(t) + N_{A\rightarrow E}(t); \end{aligned} \tag{3.1}$$

where H is the channel gain and N is the zero mean additive Gaussian noise [58]. If B responds with a signal R_B to A, then the received signals by A and E may be modeled as follows:

$$\begin{aligned} R_{B\to A}(t) &= H_{B\to A}(t) \times S_B(t) + N_{B\to A}(t); \\ R_{B\to E}(t) &= H_{B\to E}(t) \times S_B(t) + N_{B\to E}(t); \end{aligned} \tag{3.2}$$

Suppose, $S_A(t)$ and $S_B(t)$ are two probe signals, known to A, B, and E. From the received signals $R_{A\to B}(t)$, $R_{B\to A}(t)$, $R_{A\to E}(t)$, and $R_{B\to E}(t)$, the channel gains can be estimated and are denoted as $H'_{A\to B}(t)$, $H'_{B\to A}(t)$, $H'_{A\to E}(t)$, and $H'_{B\to E}(t)$, respectively. Due to the **reciprocity property** [58] of the wireless channel, if A and B send the probe signals to each other within the wireless channel's coherence time[1], one can assume that the estimated channel gain is the same for both A and B: $H'_{A\to B}(t) \approx H'_{B\to A}(t)$. However, from the eavesdropper's side, the estimated channel gains $H'_{A\to E}(t)$, $H'_{E\to A}(t)$, $H'_{B\to E}(t)$ and $H'_{E\to B}(t)$ will be independent of $H'_{A\to B}(t)$ and $H'_{B\to A}(t)$, if the eavesdropper is a few wavelengths [58] away from the legitimate wireless channel. Utilizing this concept, the channel gain ($H'_{A\to B}(t)$ and $H'_{B\to A}(t)$) may be used to extract PSK bits (see the technique in Section 4.1) for security purposes.

The wireless communication channel gain varies over time due to temporal or spatial variations in the environment. Typically, the channel may be modeled with a fast fading model or a slow fading model depending on the changing speed of the environment [48]. For automotive CPS, if there exists a velocity difference between two communicating automotive wireless nodes, the scheme uses a **fast fading model (temporal variation)**, otherwise, the scheme uses a **slow fading model (spatial variation)**.

[1] In a wireless communication system, the coherence time is the time duration over which the channel impulse response is considered to be invariant.

Fast Fading

A *Rayleigh fading channel* [48], which is suitable for modeling vehicular wireless communication [48] in an urban driving profile, is applied for the fast fading model. The *Rayleigh fading channel* models the *Doppler shift effect* [48] due to the different speeds between two communicating wireless nodes. In this model, the channel gain H should abide by the following Probability Distribution Function (PDF):

$$PDF_H(H, \sigma) = \frac{H}{\sigma^2} e^{-H^2/(2\sigma^2)} \tag{3.3}$$

where σ is an environment-related parameter. Due to the *Doppler shift effect*, H only remains constant within the coherence time [48] T_c (see the following Equation).

$$T_c \approx \frac{0.423}{f_d} \tag{3.4}$$

here, f_d is the maximum *Doppler frequency.* During automotive wireless communication between A and B, f_d may be decided by the speed difference of the two communicating vehicles ΔV_A as shown below:

$$\begin{aligned} f_d &= \frac{\Delta V}{c} f_0 \\ \Delta V &= |V_A - V_B| \end{aligned} \tag{3.5}$$

where c is the speed of light and f_0 is the carrier frequency.

This model reflects that the channel changes roughly every time interval of T_c. In other words, the higher the ΔV is, the more frequent the channel changes and the quicker a channel-based PSK may be generated. Extracting information from the channel gain H to generate a secret bit must be done within a given time period, T_c. Otherwise, the changes in the channel after T_c may cause mismatches between the generated PSKs of the communicating automotive wireless nodes.

Slow Fading

When the relative speed between the communicating automotive wireless nodes is low, $\Delta V \approx 0$, the fast fading model will not work. Therefore, the scheme instead uses a general slow fading model for the wireless communication. In a slow fading channel, the gain remains correlated in time if the channel does not move over a certain distance. This distance is defined as the coherence length L_{coher}. On the other hand, the model assumes that if the channel moves further than L_{coher}, the channel gain will become independent of the previous channel gain. Therefore, considering the velocity of the vehicle V, one may calculate the coherence time for a slow fading channel as follows:

$$T_c \approx \frac{L_{coher}}{V} \tag{3.6}$$

Similar to the fast fading channel model, the slow fading channel also changes roughly every time interval, T_c. As demonstrated with Equation 3.6, L_{coher} is decided by the environment. In other words, the higher the V is, the more frequently the channel physically changes. The time varying channel gain for a slow fading model follows the log-normal distribution as shown below:

$$PDF_H(H, \sigma) = \frac{1}{H\sigma\sqrt{2\pi}} e^{-\frac{ln(H)}{2\sigma^2}} \tag{3.7}$$

3.1.2 Attack Model

The vehicular communication attack model is based on **a classic non-intrusive wireless attack model** where the attacker tries to decipher the message by eavesdropping on packets from the legitimate wireless channel through a separate wireless channel. We assume that the attacker can capture all the wireless packets sent through the legitimate wireless channel and the attacker knows all the information about the communication system including modulation/coding techniques and cryptographic algorithms. Therefore, in such a scenario, if the attacker can get the related pre-shared or cryptographic key, the system security requirements will be broken. As a result, one may define attack strength *AttackStr* as a rate at which the attacker can employ a given amount of computing hardware resources to evaluate a number of keys within a period of time. We note that intrusive attacks are not considered in this paper since they typically require the use of highly expensive and impractical devices and are challenging to implement on specific vehicles in real-time scenarios. Further, note that no knowledge about the attacker is necessary (such as channel state information [36]) for the algorithm as the attacker will generally be farther than a wavelength (approximately 5 centimeters for the 802.11p automotive communication protocol [49]) away from the legitimate wireless channel. Notice that, in the case of V2I communication, the infrastructure is typically physically protected and makes it difficult for attackers to eavesdrop the messages within a physical distance of a few wavelengths from the infrastructure.

3.2 Vehicular Visible Light Wireless Communication

3.2.1 System Model

The proposed physical layer secret key generation method exploits the randomness in the received visible light signals due to road conditions and driving behavior. Since the frequency of vehicular VL communication is extremely high, there is no need to use a low or high fading model like those for RF. Instead, the vehicular VL communication system model is based on the vehicle trajectory data provided by NGSIM program [41], the road surface roughness, and headlight modeling [8]. Since the Lambertian model is not an accurate model to simulate the intensity pattern of a vehicles headlight and taillight, a market-weighted headlamp beam model is utilized [45]. Using the luminous intensity (candela) table provided in this model the corresponding illuminance value at any point of interest is calculated. A Line of Sight (LOS) communication is assumed. The illuminance (L) at the photodetector (PD) at the vertical angle (θ) and horizontal angle (ϕ) with respect to the headlamp axis is determined by the following equation [29],

$$L = I(\phi, \Theta)\ x\ (d\omega/dA) = I(\phi, \Theta)\ x\ (cos\tau/r^2) \tag{3.8}$$

where r,dA,$d\omega$,τ,$I(\phi, \Theta)$ are communication distance, photodetector (PD) area, solid angle, the angle between the photodetector normal and the incident direction, and luminous intensity respectively. Then the received Line of Sight (LOS) optical power ($PRX - LOS$) is calculated by $P_{RX-LOS} = (L\ x\ A_r)/LER$ when $0 \leq \tau \leq \Omega$, otherwise $P_{RX_{LOS}} = 0$ [30] where A_r, Ω, and LER are PDs total area, the half angle of PDs field of view (FOV) and the luminous efficacy of radiation, respectively. From the equation above, the received optical power is calculated, and from the optical power the photodetector current is calculated. Moreover, it is assumed that the taillight follows the same model as the headlight but with much lower intensity. The shot noise is considered due to background solar radiation and other artificial lights. Thermal noise associated with the receiver is also considered, as mentioned in [30]. Relative velocity and relative lateral and longitudinal distances among vehicles result in random yet symmetrical variation in the intensity patterns of received signals of a pair of communicating embedded devices on separate vehicles. This randomness can readily be exploited to generate symmetric cryptographic keys.

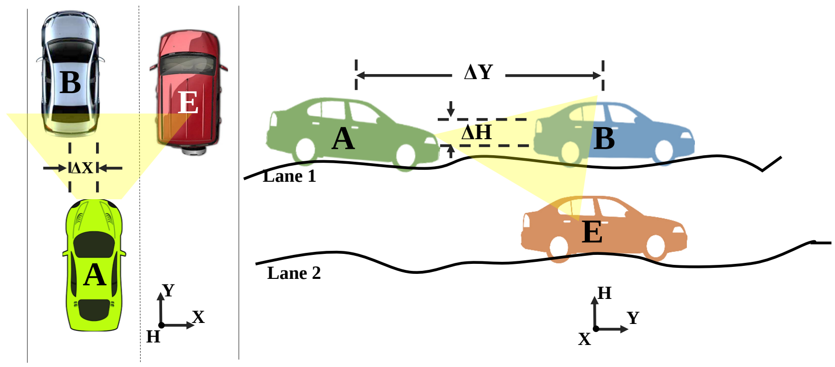


Figure 3.1: Vehicular Visible Light Communication Channel Model

3.2.2 Attack Model

To generate symmetric keys and to assess the feasibility of the key generation scheme, a model of communication links between the vehicular transceivers Alice(A) and Bob(B) and another communication link between Alice (A) and the adversary Eve (E) is developed (Figure 3.1). When Alice and Bob want to generate a symmetric key, they need to exchange a pre-defined probe signal (PRBS modulated bit pattern with a predefined length). To increase the reliability of the proposed method, the data of the vehicles such as speed, lateral coordinate (X), longitudinal coordinate (Y), time, etc. are extracted from the vast amount of data provided by the Next Generation SIMulation (NGSIM) program (under the Federal Highway Administration) [30]. Moreover, using big data analysis a combination of three vehicles (Alice, Bob and Eve) are chosen, where all three are omnipresent in the vicinity of each other in the real world with the intended point-to-point link establishment between Alice and Bob. Then from the NGSIM data, the relative lateral (ΔX) and longitudinal (ΔY) distances between selected transceivers over a time duration are calculated. These relative distances are totally stochastic (Figure 3.2). Stochastic road surface roughness (ΔH) is also added to the mathematical model [8]. From all this information a received intensity distribution can be derived at the photodetector.

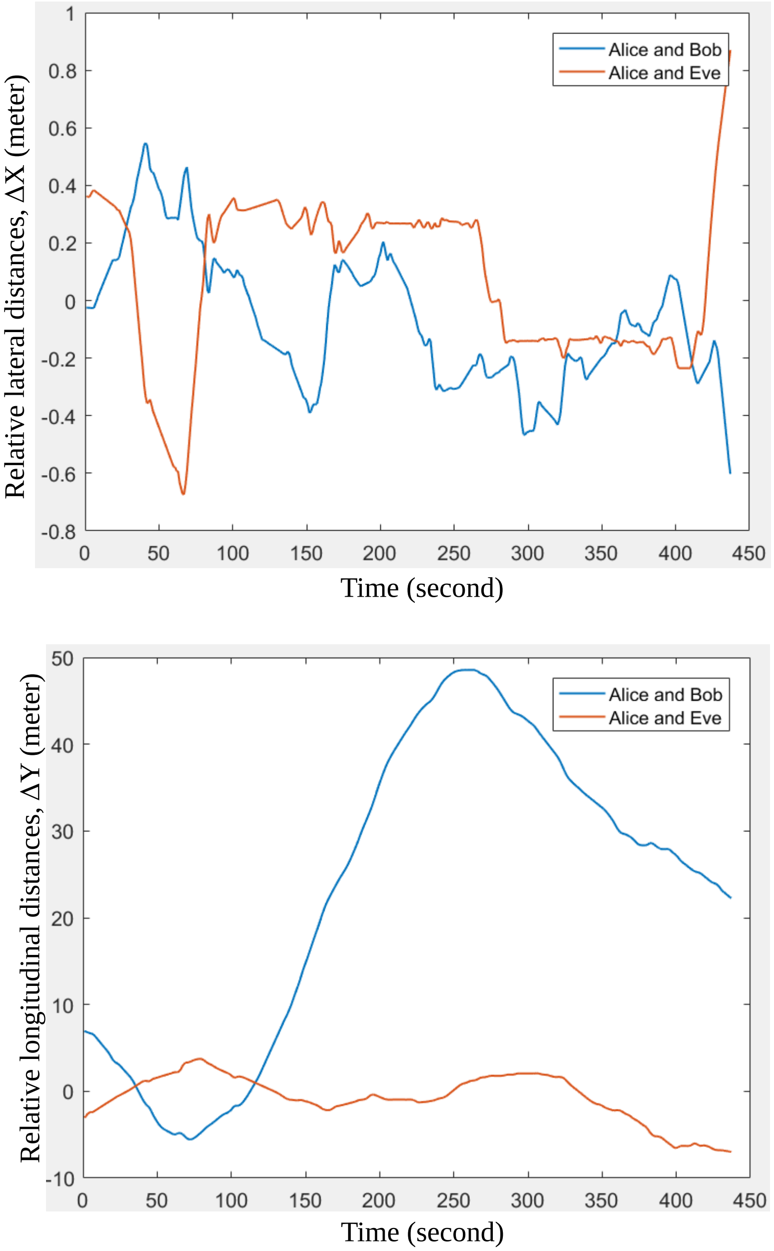


Figure 3.2: Relative Vertical and Horizontal Distances Between Communicating Vehicles

Chapter 4

Physical Layer Pre-Shared Key Management Scheme

The following sections focus on the PSK generation algorithm which can be used to help secure V2X communication. The main focus is on the vehicular RF wireless communication since it is currently more developed and understood than vehicular VL wireless communication. However, the algorithms are generic and easily customizable for use in VL wireless communication applications.

4.1 Pre-Shared Key Generation

As shown in Figure 4.1, there is a V2V wireless communication session between two vehicles. Both Alice and Bob are driving, where Alice's vehicle (A) is communicating with Bob's vehicle (B). Assume the driving velocities for A and B is V_A and V_B, respectively and the velocity difference between these two moving vehicles is ΔV. The coherence time T_c of the communication channel between A and B may be estimated using Equation 3.4 and

Equation 3.5. Now, if A and B want to generate a PSK with size of PSK_{size}, they need to exchange a set of pre-defined probe signals (can be any kind) to evaluate the randomness of the channel gain H using Equation 3.3. In order to have a low mismatch rate, they must exchange each probe signal within the Coherence Time (T_c) interval. Meanwhile, in order to keep bits of the generated PSK uncorrelated to each other, the time interval defined as τ_{step} between exchanging each probe signal should be no less than T_c. Notice that, as long as the sender A and receiver B share the same τ_{step}, the process of exchanging pre-defined signals is naturally synchronized. It is assumed that there exists a pre-defined τ_{step} for both A and B. Otherwise, since knowing τ_{step} will not help the attackers to derive the generated PSK, one suggested solution would be that the sender A and receiver B may make an agreement on τ_{step} through public communication before the PSK generation process.

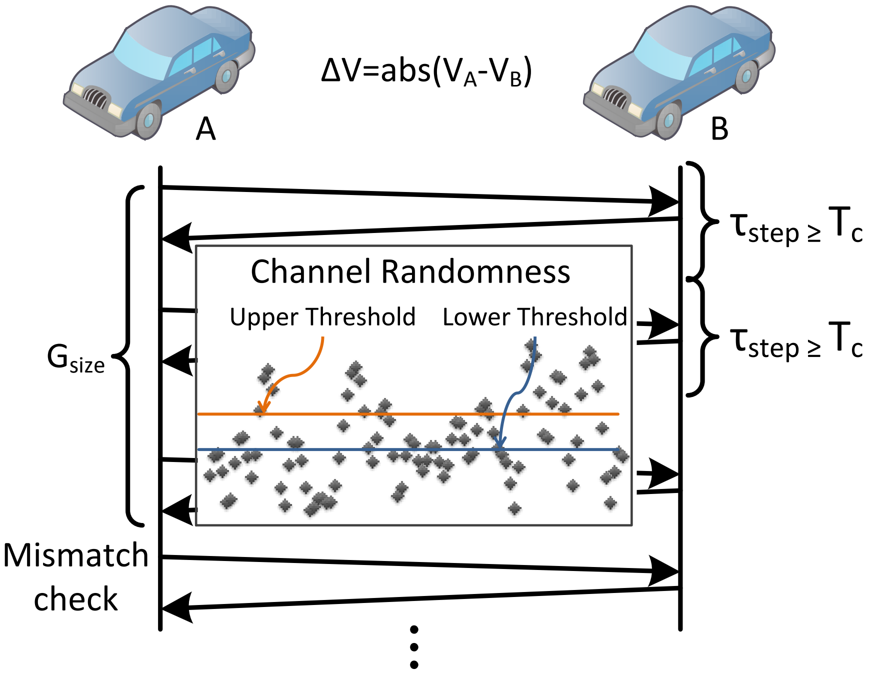


Figure 4.1: Physical Layer PSK Generation for a V2V Scenario

4.1.1 Algorithm Pseudocode and Description

A predefined group of probe signals with a group size g is sent for evaluating the channel randomness. After the probe signals are exchanged, a set of measured Received Signal Strength (RSS) values is used to generate secret key bits on each side. A mismatch checking step is also implemented to remove mismatching bits. During this step, the sender and receiver will publicly exchange the indexes of the probe signals which are used for generating secret bits, in PSK_{idx} and remove the mismatched indexes. Notice that the exchange is public and the attacker may easily get PSK_{idx}. However, the attacker will not be able to figure out the generated bits because only the sender and receiver share the RSSI values of the probe signals. Once a set of matching bits is generated, the set's size must be greater than or equal to the required PSK length, L_{PSK}. If the set of bits is not long enough, the whole process reiterates until it is. The pseudocode of the wireless channel-based PSK generation algorithm is presented in Algorithm 1.

Lines 3-5 take $(g \times \tau_{step})$ time to collect all Received Signal Strength (RSS) values from the wireless channel. Line 6 filters the low frequency parts of the collected RSS values with a high pass filter defined by its impulse frequency response $H_{highpass(t)}$. The filtered signal values $RSS_{filtered}$ contains all the information necessary to extract the secret bits. Lines 7-10 calculate the thresholds used for generating bits from the received RSS values. As proposed by [40], there are two thresholds. Every RSS value greater than the upper threshold Th_{up} is considered as a 1 bit and every RSS value less than the lower threshold Th_{lo} is considered as a 0 bit. Any value in between Th_{up} and Th_{lo} is discarded. The thresholds Th_{up} and Th_{lo} are calculated by the equations in Line 9 and Line 10, respectively, based on the mean and variance values of the collected RSSs. Additionally, there is α, which is a configurable and tuning parameter that may help reduce the bit mismatches due to the existence of noise. If the signal-to-noise ratio of the channel and the transmitters is low, α may need to be set as a higher value to reduce the number of mismatches; otherwise, α may be chosen to be a lower

Algorithm 1: A Wireless Channel-Based PSK Generation Algorithm for Automotive CPS

```
Input: Measured Signal Strength: $RSS$
Input: Sample Time Step: $\tau_{step}$
Input: Group Size: $g$
Input: Threshold Parameter: $\alpha$
Input: Required Pre-Shared Key Length: $L_{PSK}$
Output: Generated Pre-Shared Key: $PSK$
$L = 0$; $PSK = 0$; $RSS_{set}$=$\emptyset$; $RSS_{filtered}$=$\emptyset$; $PSK_{idx}$=$\emptyset$;
while $L < L_{PSK}$ do
    for $i$=1 to $g$ do
        $RSS_{set} = RSS_{set} \cup RSS_i$;
        Wait($\tau_{step}$);
    $RSS_{filtered} = RSS_{set} * H_{highpass}(t)$;
    $MeanValue$ = Average of $RSS_{filtered}$;
    $Var$ = Variance of $RSS_{filtered}$;
    $Th_{up} = MeanValue + \alpha * Var$;
    $Th_{lo} = MeanValue - \alpha * Var$;
    foreach $RSS_j \in RSS_{filtered}$ do
        if $RSS_j > Th_{up}$ then
            $PSK = (PSK << 1) + 0$;
            $L = L + 1$;
            Record $j$ in $PSK_{idx}$;
        else if $RSS_j < Th_{lo}$ then
            $PSK = (PSK << 1) + 1$;
            $L = L + 1$;
            Record $j$ in $PSK_{idx}$;
    Exchange $PSK_{idx}$;
    Remove mismatch bits from $PSK$;
return $PSK$;
```

value to improve the performance of the algorithm. Lines 11-19 check all the collected RSSs and generate a PSK, PSK, with length, L. Notice that, Line 15 and Line 19 also record the index of all suitable RSSs for generating secret bits. The indexes in PSK_{idx} from the two communicating automotive wireless nodes are exchanged in Line 20. Then, in Line 21, PSK_{idx} is used to remove all mismatching bits. Finally, if $L >= L_{PSK}$, a PSK is generated among both communicating parties; otherwise, the algorithm will reiterate.

4.2 Pre-Shared Key Length Optimization

In order to create an efficient scheme to secure wireless communication for the automotive domain an optimal PSK length for Algorithm 1 must be derived. The optimal PSK length is dependent on the V2X scenario, where a scenario is defined as an instance of communication between an automotive cyber-physical system and another entity (either automotive or non-automotive) within a specific physical setting (e.g., street or highway). For example, a scenario can be a vehicle communicating with a tolling device in a highway. In this scenario, the vehicle and tolling device will need to generate a PSK within a small amount of time [46, 14]. For these reasons, it is necessary to generate an optimal PSK length for the proposed technique to finish under timing constraints with as much security strength (which is formally defined in Section 5.1.4) as possible. More specifically, the following characteristics determine a scenario: **1)** Types of communicating parties, **2)** Location, **3)** Fading model (slow or fast) and **4)** Coherence length or time. From these details, one may determine the scenario and key lifetime to determine the optimal PSK length for the proposed physical key generation technique. For the following sections, several scenarios will be defined and used as motivating examples. These can be altered and extended according to designers.

4.2.1 Scenario Mapping

V2V Scenarios

In V2V, if a scenario is an emergency, a communication session would last less than a second, but if it is not, a session could last to many minutes for traffic efficiency purposes. Therefore, a V2V scenario is based on the severity of the situation as summarized by the following: **1)** Emergency Avoidance (milliseconds), **2)** Emergency Detection (seconds) and **3)** Traffic Efficiency (minutes) [46]. It can be possible to evaluate the severity of a V2V scenario by the amount of time that has passed since the start of the communication session. To do this, it is assumed that all initiated V2V communication sessions are first categorized as Emergency Avoidance to prevent unexpected collisions. After many key refreshes, the scenario time constraints can be relaxed to generate a longer PSK for higher security strength and longer lifetime.

V2I Scenarios

V2I communication scenarios include communication with roadside units, tolling stations, and traffic lights. Infrastructure is generally motionless, implying that computing the maximum time of communication, or total latency, requires knowledge of the overall distance that the vehicle will need to cover before it goes out of the infrastructure's communication range. In general, for V2I, the highest ideal range for realistic communication is approximately .5 miles (750 meters) to .6 miles (1000 meters) [49, 5, 14].

Computing the optimal PSK length for a scenario therefore requires us to calculate the total latency (see next subsection) and the coherence times corresponding to all possible relative velocities. Although this work assumes relative velocity ranges such as 0-45 Miles per Hour (mph) for a street scenario and 0-75 mph for a highway scenario [49], it is important to note that the actual relative velocity is computed before PSK length optimization by the vehicles' embedded devices.

Total Latency

Total latency is the assumed limit to how long two entities in vehicular communication will communicate. For V2V scenarios, the total latency of communication between two vehicles is set to be 200 milliseconds [46], since an Emergancy Avoidance scenario is first assumed. On the other hand, the V2I scenarios are subdivided according to the setting (e.g., street or highway), where the relative velocities in V2I highway scenarios are (in general) greater than those in V2I street scenarios. By considering the maximum possible velocities for each type of scenario, the highway and street communication total latencies are approximately 10 seconds and 40 seconds, respectively.

Algorithm 2: Scenario Mapping

```
Input: Scenario: $Scen$
Input: Fading Model: $Model$
Output: Coherence Times: $CoherTimes$
Output: Total Latency: $TotalLat$
$CoherTimes = \emptyset$
$TotalLat$ = getTotalLatency($Scen$)
if $Model == FastFadingModel$ then
    $VelocityDiffs$ = getVelocityDifferences($Scen$)
    foreach $VelocityDiff_i \in VelocityDiffs$ do
        Compute coherence time, $CoherTime_i$ using $VelocityDiff_i$
        $CoherTimes = CoherTimes \cup CoherTime_i$
if $Model == SlowFadingModel$ then
    $Velocities$ = getVelocities($Scen$)
    foreach $Velocities_i \in Velocities$ do
        Compute coherence length, $CoherLength_i$ using $AsphIndex$, $SpeedOfLight$,
          and $Bandwidth$
        Compute coherence time, $CoherTime_i$ using $Velocity_i$ and $CoherLength_i$
        $CoherTimes = CoherTimes \cup CoherTime_i$
return $CoherTimes$, $TotalLat$
```

Fading Models

The decision of the fading model is important in the proposed scheme. It is reasonably assumed that a modern automotive system includes various sensors, such as a speed radar, to provide an estimation of the relative speed between other vehicles. As a result, a possible fading model selection solution is described in the following:

Fast Fading: For velocity differences greater than 5 mph, the fast fading model is used for the scenario mapping technique. From the range of velocity differences, corresponding coherence times are computed using equations 3.4 and 3.5 in Section 3.1.1.

Slow Fading: For low velocity differences such as 5 mph or under, the slow fading model is required to compute the coherence times and corresponding optimal PSK lengths. For two unmoving communicating parties or an emergency avoidance scenario where the parties have low relative velocity (e.g., approximately 0-5 mph), it would be recommended to use a stored pre-distributed key to implement the symmetric encryption algorithm instead.

When the proposed algorithm uses the slow fading model, equation 3.6 provided in Section 3.1.1 is used to calculate the coherence times from the coherence lengths and scenario mapping values. To compute the coherence lengths, the index of refraction of asphalt (1.635), speed of light, and bandwidth of the channel (5.9 Ghz) [49] are used. The pseudocode of the scenario mapping algorithm is given in Algorithm 2.

It is important to note that the fading model itself is not the contribution of this paper. It can be dynamically updated using more detailed models to improve the key generation performance. And the decision of the fading model can also be agreed between two communication nodes through public communication before the key generation process.

4.2.2 Algorithm Pseudocode and Description

Using the scenario mapping function and attack model, the optimal PSK length can be computed. For safety-critical and resource-limited devices, the following timing constraint is used: $LifeTime$ (the total scenario-based latency, and also the time until a new PSK must be generated to prevent an attacker from computing the previous PSK, as described in Section 4.2.1). Another parameter $Fract$, is a dimensionless input which provides the designer an option to adjust the timing constraints based on their unique requirements, such as preventing key generation from taking away valuable communication time (since key generation requires packet exchanges). For the purpose of evaluation, this is set it to be .2 but is increased to .4 for 5-10 mph and .7 for 0-5 mph. As an example, with $Fract = .2$, the algorithm will compute an optimal length such that the estimated key generation time will be within 1/5th of the lifetime. Consequently, the algorithm will ascertain out of a set of PSK lengths, the most viable optimal length that satisfies the time constraint, $LifeTime * Fract$. Since it is likely that a chosen PSK length may not meet the length requirement of the used encryption method (e.g., AES), the PSK must be extended or transformed into valid cryptographic keys through methods discussed in Section 4.3. The PSK length optimization algorithm is detailed in Algorithm 3 and simulation results are provided in Section 5.1.

In the algorithm, a simple Binary Search-based method discovers the optimal PSK length in terms of generation time under lifetime constraints. Thus, minimum and maximum lengths in terms of bits (which are defined as 1 and 128, respectively) must be defined to set the range of the search space. As the velocity difference (or velocity for slow fading model) increases, one may expect that the optimal length also increases. This is because a higher velocity difference (velocity) directly enables a higher PSK bit generation rate. For more generation time $LifeTime * Fract$, the optimal length also increases. In addition to determining the optimal PSK length, the algorithm also produces effective and generation times and energy overhead values. Notably, this algorithm is customizable and may be customized to compute the

Algorithm 3: Algorithm for PSK Length Optimization

```
Input: Minimum Length: $PSKLenMin$
Input: Maximum Length: $PSKLenMax$
Input: Power to Generate a PSK Bit: $Power$
Input: Attack Strength (128-Bit Keys per Second): $AttackStr$
Input: Scenario: $Scen$
Input: Fraction of the Lifetime: $Fract$
Input: Fading Model: $Model$
Output: PSK Generation Energy Values: $EnergyVals$
Output: Optimal PSK Lengths: $OptPSKLens$
Output: Optimal Lifetimes: $OptLifeTimes$
Output: Optimal PSK Generation Times: $OptPSKGenTimes$
Current PSK Length: $CurrPSKLen$=0
$\{CoherTimes, TotalLat\}$ = ScenarioMapping($Scen$, $Model$)
foreach $CoherTime_i \in CoherTimes$ do
    Perform Binary Search to find optimal PSK length
    while ($PSKLenMax >= PSKLenMin$) do
        $CurrPSKLen$ = Mid($PSKLenMin$, $PSKLenMax$)
        $PSKGenTime = CoherTime_i * CurrPSKLen$
        $PSKGenEnergy = Power * PSKGenTime$
        $LifeTime = \min(2^{(128-CurrPSKLen)}/AttackStr, TotalLat)$
        if $KeyGenTime <= (LifeTime * Fract)$ then
            $OptPSKLen = CurrPSKLen$
            $OptLifeTime = LifeTime$
            $OptPSKGenTime = PSKGenTime$
            $Energy = PSKGenEnergy$
            $PSKLenMin = CurrPSKLen$
        else
            $PSKLenMax = CurrPSKLen$
    $OptPSKLens = OptPSKLens \cup CurrPSKLen$
    $OptLifeTimes = OptLifeTimes \cup LifeTime$
    $OptPSKGenTimes = OptPSKGenTimes \cup PSKGenTime$
    $EnergyVals = EnergyVals \cup Energy$
return $OptPSKLens$, $OptLifeTimes$, $OptKeyGenTimes$, $EnergyVals$
```

optimal PSK length under energy constraints (which can be provided as an extra constraint in Line 16). In general, the energy constraints can be determined based on the power consumption of the vehicle's embedded device (s) when computing a single PSK bit. The optimal PSK length will serve as input for the physical layer PSK generation technique (Algorithm 1). Then a cryptographic key may be derived from it, as shown in the following section.

4.3 Cryptographic Key Derivation

For a key to be usable in cryptography, it must have length in accordance to a valid existing encryption scheme. For V2X communication, one may assume that the encryption scheme is AES-128, a fast and efficient standard symmetric encryption scheme (on the order of 9 microseconds to encrypt and decrypt 60 bytes) [46]. For AES-128, the key length must be 128 bits; however, since the primary concern is with safety-critical V2X applications, it is necessary to provide the proposed key generation algorithm with an optimized PSK length to minimize any overhead from the key management scheme. Unfortunately, doing this means that the length of the PSK may be less than 128 bits and therefore requires an additional, yet light, key derivation step to convert the PSK into an appropriate cryptographic key.

Two possible key derivation solutions can be used: **1)** the HMAC Key Derivation Function (HKDF) [24] or **2)** the Merkle-Damgard bit padding algorithm [17]. Given the key length and the total latency, one method to choose between the key derivation functions is provided in Algorithm 4. As input, the HKDF takes in the desired key length, $KeyLen$, the pre-shared key, PSK, and some mutual data as the salt, $Salt$, such as past traffic information (e.g., environmental, lane changes, speed changes). The HKDF outputs a cryptographic key with $KeyLen$ bits. If there is no prior data exchanged between each other, the two parties can use null as the Salt. The resulting key $CryptoKey$ is created by the HKDF which concatenates partial results from a one-way hash function, such as the HMAC-SHA256, on both of the inputs. Therefore, the HKDF function is: $CryptoKey = HKDF(PSK, Data, KeyLen)$. On the other hand, it is possible to implement the Merkle-Damgard bit padding algorithm on a PSK with insufficient length to convert it into a 128-bit key. The Merkle-Damgard algorithm pads a 1, then successive 0s, and finally the length of the original PSK to the end such that the length is equal to a desired length. This method is simpler and faster than the HKDF, although it does not necessarily produce strong cryptographic keys.

Algorithm 4: Algorithm for Cryptographic Key Derivation

```
Input: Scenario: $Scen$
Input: PSK: $PSK$
Input: Salt: $Salt$
Input: Desired Key Length: $KeyLen$
Output: Cryptographic Key: $CryptoKey$
if $SizeOf(PSK) < 128$ and $getTotalLatency(Scen) == .2$ then
    $CryptoKey$ = Merkle-Damgard($PSK$, $KeyLen$)
else if $SizeOf(PSK) < 128$ && $getTotalLatency(Scen) > .2$ then
    $CryptoKey$ = HKDF($PSK$, $Salt$, $KeyLen$))
else
    $CryptoKey$ = Truncate(128,$PSK$)
return $CryptoKey$
```

Notice that the short key is only used for a short session where 1) the expiration times of a message are small, 2) the key generation algorithms are typically not suitable, and 3) confidentiality is not required but integrity is. Despite the short seed length, the key refresh rate derived from the key optimization algorithm (see Algorithm 3) will help prevent attackers from easily computing the key while it is being used. By the time the attacker figures out the key and decrypts the message after it has been sent, the communication scenario may have changed considerably (for example, broadcasting of real-time traffic information, and V2V communication for emergency purposes) and the decrypted messages are no longer valid afterwards (since they typically have expiration times associated with them). Further, it is important to note that for scenarios where the timing requirements are not strict or the communication session is relatively long, key with longer lengths may be generated within acceptable time. When there is a scenario where the timing requirement is strict and confidentiality is required, keys with suitable length may be generated using another existing key generation technique [2]. This and traditional approaches can coexist with one another on the automotive system. In cases where other approaches lack in performance, this approach can be used, and vice-versa. Thus, by introducing this approach, automotive designers can design a new type of automotive system such that security is optimized in terms of both performance and security strength.

Chapter 5

Simulation and Experimentation

5.1 Radio Frequency Vehicular Communication

5.1.1 Pre-Shared Key Generation Without Optimization

In this section wireless channel-based PSK generation technique is evaluated without optimization, implying that generating a PSK is equivalent to generating the cryptographic key. MATLAB [32] was used to simulate the automotive wireless channel with the following parameters. The average driving speed is set to 37 mph and the coherence length for slow fading is set to 20 meters, or about .01 miles, for an urban environment [56]. The simulation evaluates the key generation time with respect to the relative speed between two communicating nodes (0 to 75 mph in the setup). Moreover, the simulation is conducted with respect to 6 different key sizes (56, 112, 128, 168, 192, 256 bits) proposed by the security standards from NIST [3]. The summarized simulation setup is presented in Table 5.1.

As presented in Figure 5.1, the key generation algorithm has negligible performance (10 to 100 milliseconds) overhead when the relative speed is high due to the fast fading of the

Table 5.1: Experimental Setup For the Key Generation Algorithm

Tested Key Length (bits)	56, 112, 128, 168, 192, 256
Relative Velocity Range (mph)	0 to 75
Average Velocity (mph)	37
Signal to Noise Ratio (dB)	80
Coherence Length (mi)	.01
Group Size (bits)	10

wireless channel. This implies that the algorithm can be suitable for various V2X applications and scenarios. On the other hand, for the scenario where the relative speed between two nodes is around zero such as intra vehicle communication, the simulation results show a longer generation time (around 1 to 2 minutes). However, compared to the lifetime of the key, which is typically several hours to even months in these scenarios, several minutes can also be considered as negligible. Although in some cases, several seconds of overhead for generation is not acceptable (e.g., safety related applications), the wireless channel-based key generation algorithm can be optimized using the schemes in Section 4.2.

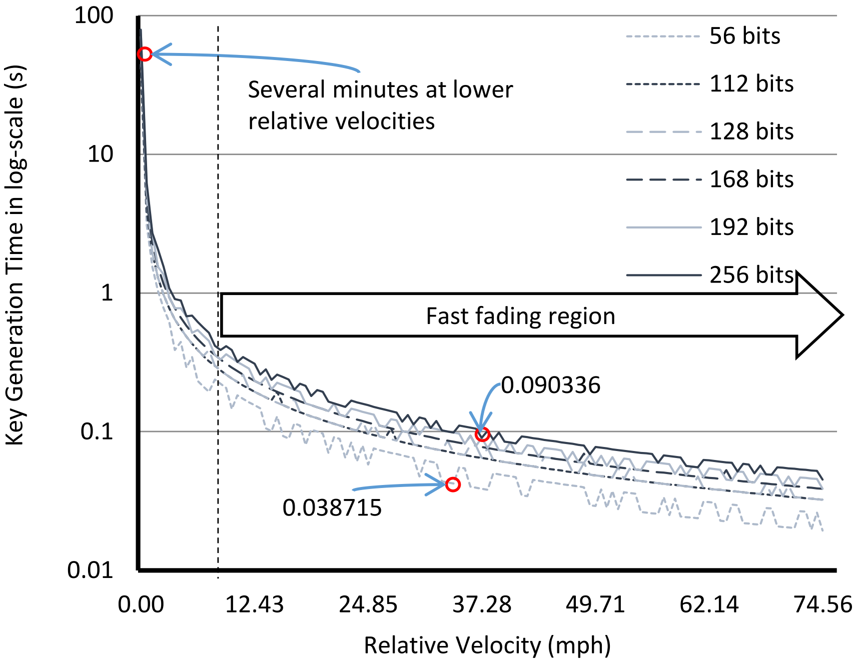


Figure 5.1: Simulation Results of Key Generation Overhead

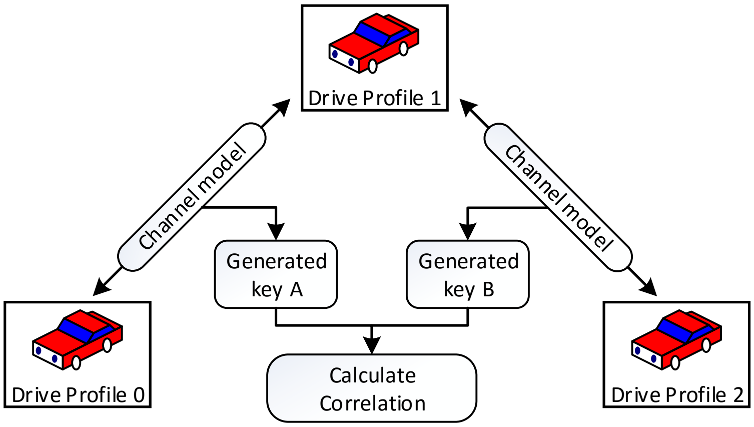
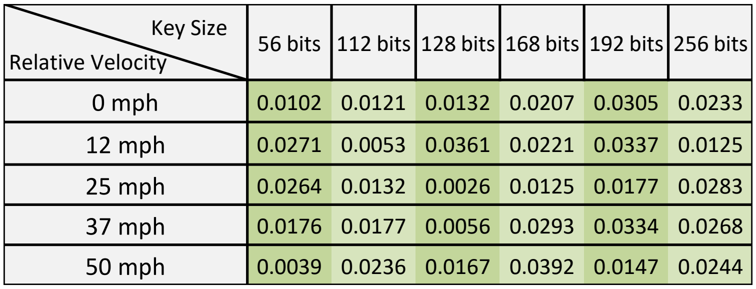


Figure 5.2: Simulation of Generating Two Secret Keys at the Same Time

Additionally, simulations were conducted to confirm the independence of two generated keys from two different automotive wireless communication channels to demonstrate that the attacker cannot easily retrieve the key by eavesdropping. The simulation setup is presented in Figure 5.2. Three vehicles (with driving profiles) are modeled and connected using the developed wireless channel models. Two wireless channel models are instantiated in the simulation, where one connects the vehicle models with *Drive Profile 1* and *Drive Profile 0* to each other, and the other connects the vehicle models with *Drive Profile 1* and *Drive Profile 2* with each other.

For each relative speed in mph and key size, as specified in Table 5.2, simulation was ran 100 times to generate two vectors of keys from two wireless channels at the same time. Then, the Pearson's correlation coefficient [26] between these two vectors was calculated. The calculated correlation results are presented in Table 5.2. From the simulation results, one may observe that all the correlation results are close to zero (the highest correlation value is just 0.0392). These results demonstrate the low correlation of keys generated from the channels of two vehicles connected to the same target through wireless communication, **thus implying that the attacker cannot retrieve the key generated from the legitimate wireless channel by this method**.

Table 5.2: Correlations of the Generated Keys



Key Size Relative Velocity	56 bits	112 bits	128 bits	168 bits	192 bits	256 bits
0 mph	0.0102	0.0121	0.0132	0.0207	0.0305	0.0233
12 mph	0.0271	0.0053	0.0361	0.0221	0.0337	0.0125
25 mph	0.0264	0.0132	0.0026	0.0125	0.0177	0.0283
37 mph	0.0176	0.0177	0.0056	0.0293	0.0334	0.0268
50 mph	0.0039	0.0236	0.0167	0.0392	0.0147	0.0244

5.1.2 Pre-Shared Key Length Optimization

In this section, PSK length optimization algorithm is evaluated with four example scenarios: V2V Street, V2V Highway, V2I Street and V2I Highway. Appropriate total latencies, velocity differences or velocities by using the suitable fading models, maximum velocities and incremental values are generated by the scenario mapping algorithm (Algorithm 2). Furthermore, from these data, the coherences and lifetimes are computed, assuming an attacker with low budget and corresponding strength of $2.3 * 10^7$ (128-bit keys per second) [Ecrypt ii]. Finally, optimal PSK lengths for each scenario are calculated. The results are shown in Figures 5.3 and 5.4 and optimization parameters are provided in Table 5.3.

Additionally, as shown in Figures 5.5 and 5.6 the PSK generation time is provided in accordance to optimized PSK lengths. From the figures, it is quite apparent that higher velocity difference implies a lower key generation time. For V2I scenarios, the optimal PSK length of 128 bits can be generated within milliseconds to seconds. As for V2V scenarios, where time is of the essence to detect and prevent collisions, it is apparent that the key generation times must be lower in comparison (up to 100 times smaller) to those of the V2I scenarios. Nonetheless, it is possible to create a PSK with 5-50 bits and convert it using the cryptographic key derivation techniques specified in Section 4.3.

Table 5.3: Scenario Mapping Data

Scenario	Total Latency (s)	Velocity Range (mph)	Increment (mph)
V2V Streets	0.2	0-45	3
V2V Highway	0.2	0-75	5
V2I Streets	40	0-45	3
V2V Highway	10	0-75	5

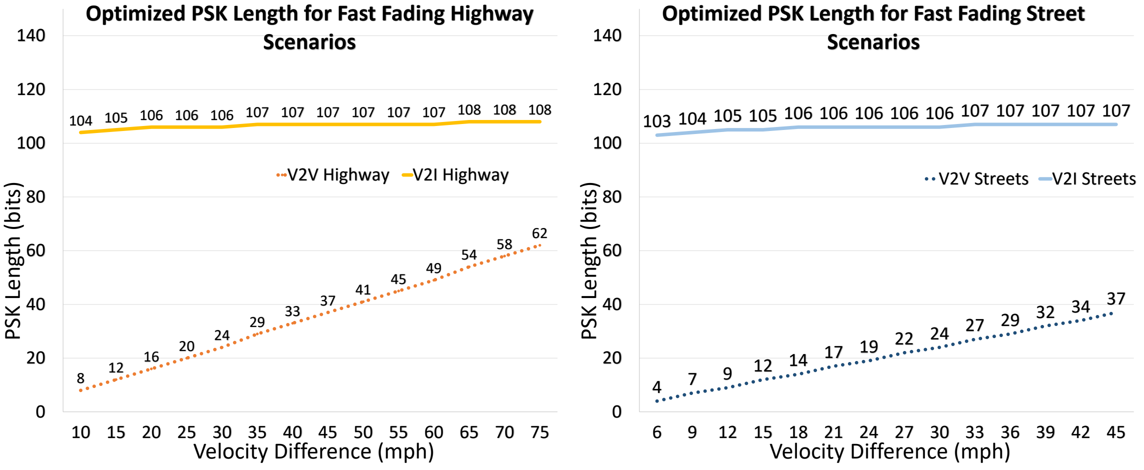


Figure 5.3: Optimized Key Length for Fast Fading Model

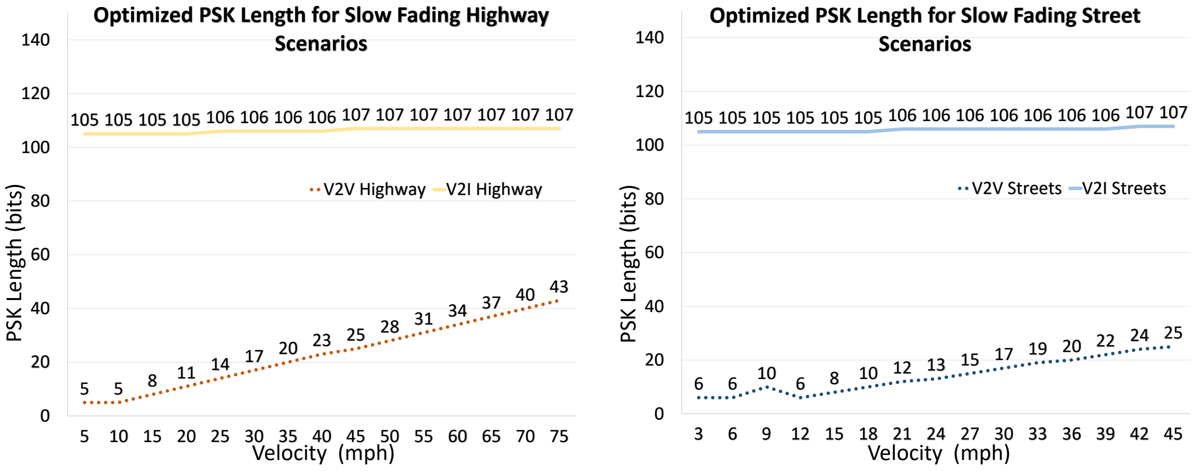


Figure 5.4: Optimized Key Length for Slow Fading Model

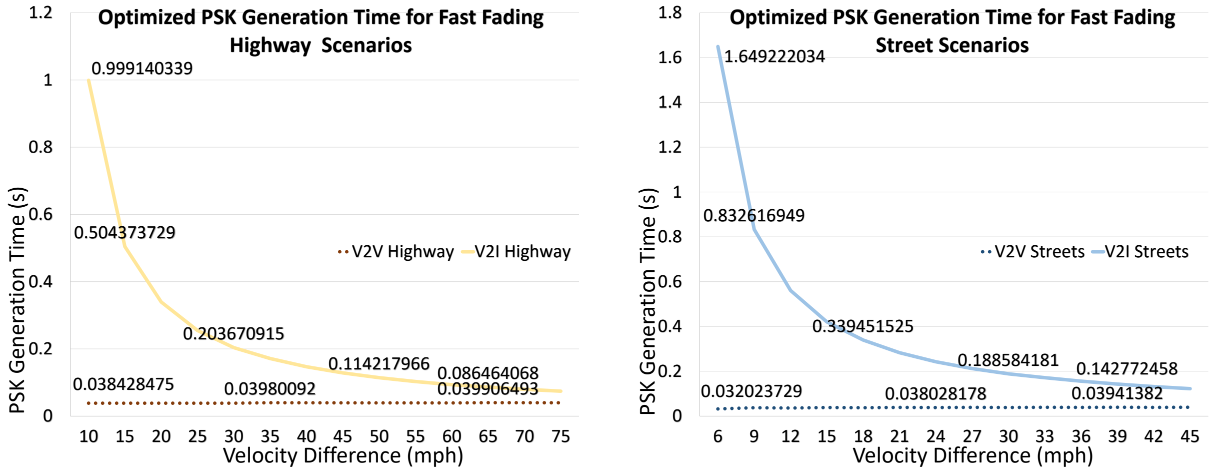


Figure 5.5: Optimized Key Generation Time for Fast Fading Model

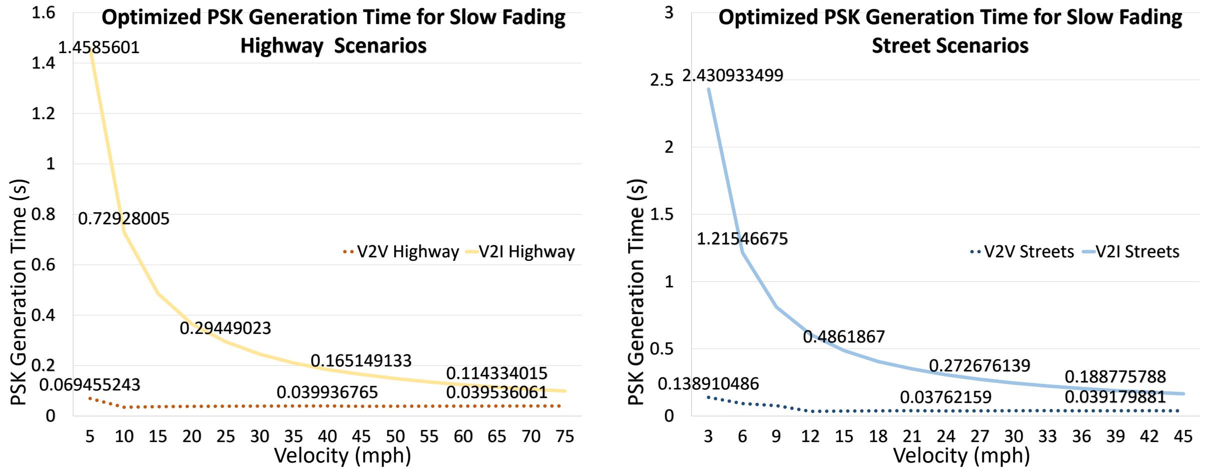


Figure 5.6: Optimized Key Generation Time for Slow Fading Model

5.1.3 Experimental Results

Going further than simulation, real world experiments are conducted to validate the proposed physical layer key generation technique. In this section, **the PSK is treated as the cryptographic key to evaluate and demonstrate the practicality of the solution using the physical layer key generation algorithm (Algorithm 1).**

Remotely Controlled Car Environment

The first experiment involved a system made up of three Remotely-Controlled (RC) cars and Raspberry Pis connected via Bluetooth. As presented in Figure 5.7, the Raspberry Pi systems are mounted on top of the RC cars. On each Raspberry Pi board, Bluetooth dongles (via USB) are attached to establish the wireless communication. One of the primary objectives is to confirm nearly zero correlation between generated keys from different channels within a short distance, but longer than a few wavelengths (for Bluetooth, the wavelength is approx .125 meters). Therefore, two Bluetooth dongles were mounted on *Car 1* (as shown in Figure 5.7) to establish two wireless communication channels between *Car 1* and *Car 0*, and *Car 1* and *Car 2*. During runtime, all the Received Signal Strength (RSS) values from each Bluetooth dongle were collected by a computer through separate Wi-Fi channels (as shown in Figure 5.7). Thus, in total there were four sets of RSS values collected from all Bluetooth dongles. Although for this experiment a computer was used to execute the key generation algorithm and analyze its results, one may easily implement the key generation algorithm on the Raspberry Pis.

The experimental environment with RC cars is considered as a slow fading one because the cars move at low speeds (less than 5 mph) and within a distance of 10 meters (about .006 miles) from each other in open areas with few moving objects around them. 200 samples of the collected RSS values are presented in Figure 5.8. From the results, one may easily observe that the RSS values collected at *Car 1* and *Car 0* for the wireless communication between *Car 1* and *Car 0* are highly correlated with each other (shown in red lines), a good sign. The same results are also found for the wireless communication between *Car 1* and *Car 2* (shown in blue lines). These results clearly show the reciprocity characteristic of the wireless communication channel. Moreover, despite the short distance between the two wireless channels, the generated RSS values from the two different wireless communication channels have nearly zero correlation, thus supporting the assumption that “an attacker that

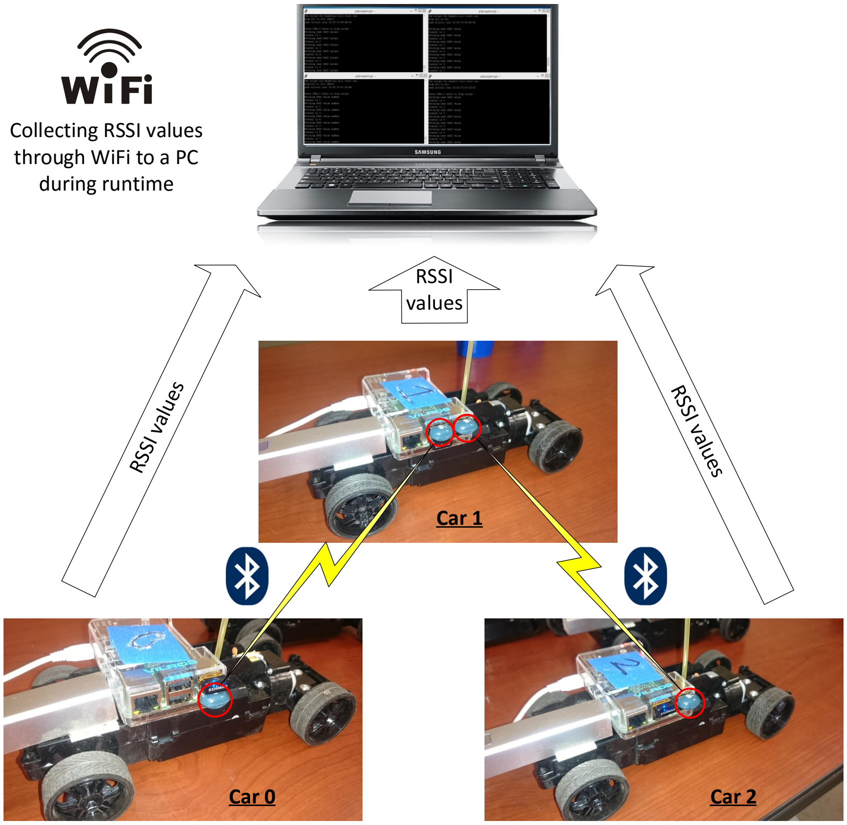


Figure 5.7: RC Car Experiments Setup

is **several wavelengths** away will experience different wireless channel characteristics, and therefore cannot obtain or predict the keys." Table 5.4 shows the generated 64 bits of keys based on the collected 200 samples of data. Notice that, 50 is used as the probe signal group size g for the key generation algorithm in this experiment.

Table 5.4: Generated 64-bit Keys from the RSS Values

	Generated 64-Bit Keys
Car 1 from Car 0	1100000110000000_0000000100000110_ 0000000010000000_0000011111111111
Car 0 from Car 1	1100000110000000_0000000100000110_ 0000000010000000_0000011111111111
Car 1 from Car 2	0000001111111111_1111000000000000_ 0000011111100000_0000011110000011
Car 2 from Car 1	0000001111111111_1111000000000000_ 0000011111100000_0000011110000011

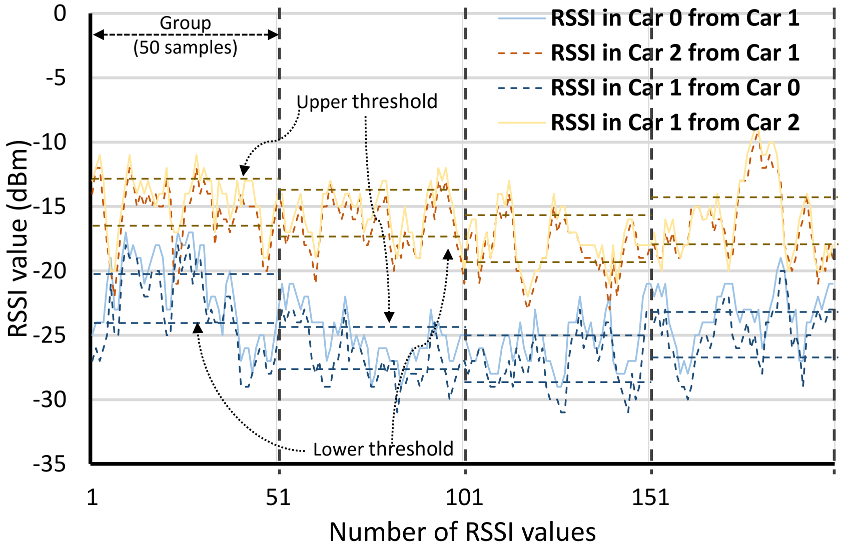


Figure 5.8: Collected Samples of RSS Values from the RC Car Experiments

Real Automotive Environment

In order to further validate the practicality of the PSK generation algorithm, experiments in real driving scenarios were also performed. To do so, Bluetooth was once again applied in customized applications to acquire RSS values of the wireless channel in real time.

As presented in Figure 5.9, mobile devices in two vehicles were placed and used to record the RSS values to generate keys. The RSS values received from both sides of the mobile devices during a period are provided in Figure 5.10. One may observe that there exists several mismatched signals in Figure 5.10, this is primarily because during that interval of time, the Bluetooth communication was not stable between the phones in the two moving vehicles, thus resulting in some loss of RSS data. However, Algorithm 1 already considers these mismatches and handles them well. In this experiment, notice that the RSS value sampling period is 10 milliseconds due to the limitations of the Bluetooth devices (mobile phone and laptop in this experiment). Due to the low sampling rate, each RSS sample will always be obtained after each coherence time, thus prohibiting the algorithm from reaching its full speed.

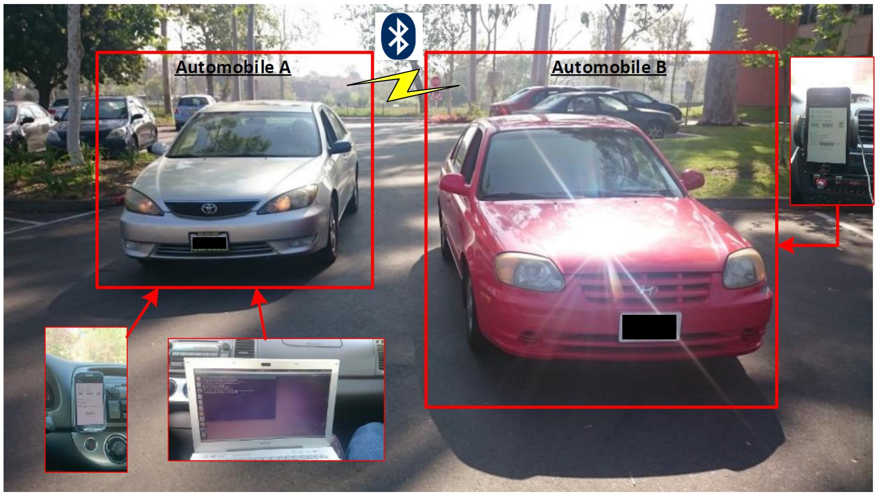


Figure 5.9: Real World Experiment Using Phones and Laptops

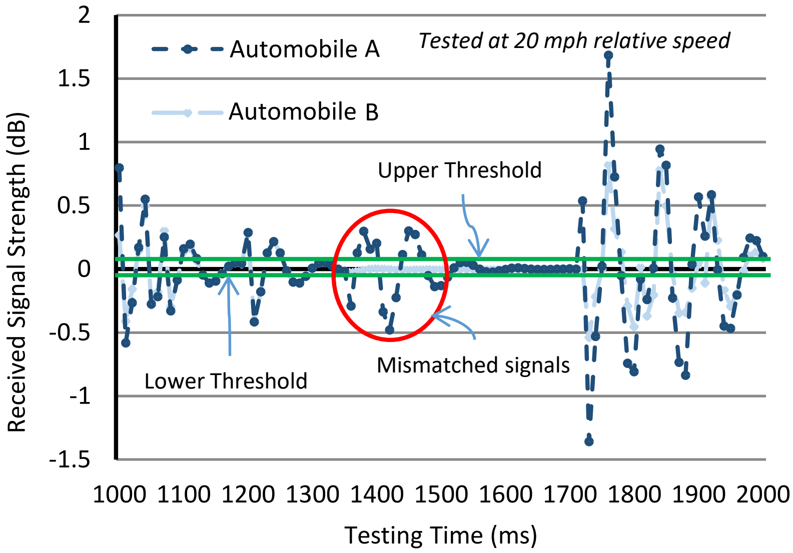


Figure 5.10: Collected RSS Values from Real World Experiment

The experiments are conducted based on three relative speeds of 20, 10, and 2 mph with both vehicles moving in the same direction. Throughout the experiment, the RSS values and generated six keys with unique lengths (see Figure 5.11). Notice that sampling takes the majority of time and the key generation algorithm's execution time is negligible (constant).

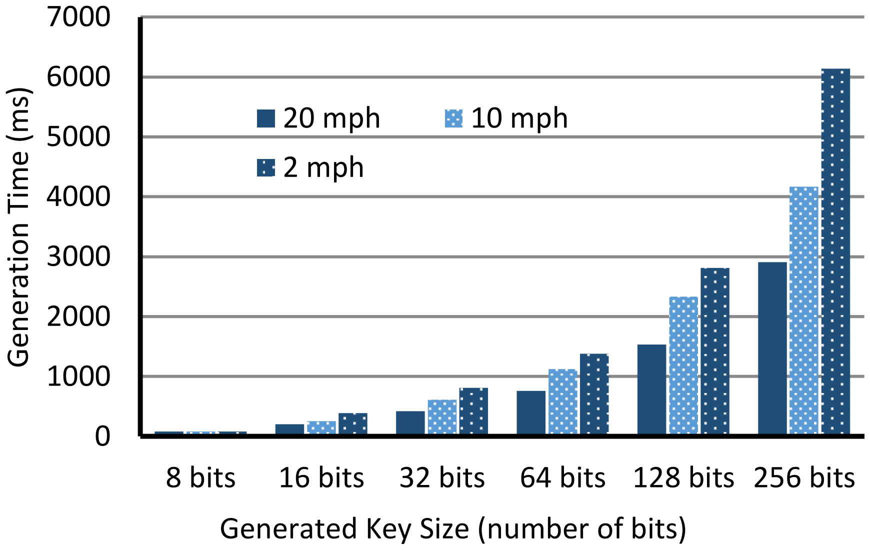


Figure 5.11: Experimental Key Generation Time at Different Speeds

5.1.4 Evaluation

In this section, the proposed physical layer PSK generation algorithm is compared with state-of-the-art hybrid cryptographic algorithms [35, 46] in terms of security strength, performance and code size overhead for automotive wireless communications.

Security Strength

Security strength indicates the amount of work (number of operations) an attacker would need to do in order to break a cryptographic algorithm or system. According to the National Institute of Standards and Technology (NIST) standard, an algorithm or system is defined to have "N-bits security strength" when it requires an attacker to perform around 2^N operations to break the algorithm or system [2]. The security strength is defined as the number of bits in the PSK, which is the basis for the cryptographic key.

Nonetheless, in order to directly measure the randomness and security strength of the PSK, in this thesis the concept of min-entropy [21] is used. As a worst case estimation, min-entropy provides the lower bound of randomness. Let K be the set of all possible PSKs randomly generated, the min-entropy is defined as follows:

$$H_\infty = H_{min} = -log(\max_{k \in K} Pr[K = k]) \tag{5.1}$$

where, $Pr[K = k]$ is the probability of generating PSK $k \in K$.

Thus, the security strength, $Security_{str}$, of a cryptographic algorithm or system is modeled using the average min-entropy on each bit of the key as follows:

$$Security_{str} = H_{min}/Key_{size} \tag{5.2}$$

where, Key_{size} is the size of the key and $Security_{str}$ is a value ranged from 0 *to* 1 in the unit of bits. For example, a 128-bit key with $Security_{str} = 0.5$ *bit* will have 64 *bits* of min-entropy.

Security Comparison

In this section, the security strength of the algorithm's generated keys are compared to those produced by other techniques. The security strength is evaluated and compared by using the proposed average min-entropy as the Key Performance Indicator (KPI). Traditional wireless sensor communication uses pre-distributed keys [37] for their practicality (a simple key management scheme) in achieving real-time communication. However, since the pre-distributed keys and associated algorithms are predictable, the pre-distributed key approaches have little to no entropy [37]. In comparison to the traditional approach, approaches that use PUFs [1], such as the SRAM-PUF [21], can generate keys with high average min-entropy.

To estimate the average min-entropy of the key generation algorithm, the algorithm was

[1] A Physical Unclonable Function (PUF) is a function based on physical characteristics that are practically impossible to be duplicated.

simulated 12800 times to generate $100 * 2^8 = 12800$ 8-bit PSKs. Based on these keys, the probability Pr_{max} of the key with the highest likelihood and applied this Pr_{max} to Equation 5.1. Figure 5.12 shows the resulting average min-entropy of the technique in comparison with other well-known techniques such as pre-distributed keys, Latch-PUFs, DFF-PUFs, and SRAM-PUFs. Note that the algorithm can generate keys with security strength close to that of some of the best PUF-based approaches (up to 67% average min-entropy for 8-bit keys[2]). Although some of the PUF-based approaches (e.g., SRAM-PUF) can generate keys with higher average min-entropy (since the number of 0 and 1 bits tend to be around the same), the algorithm has the advantage of generating keys by directly accessing the communication channel without needing a special physical process such as SRAM rebooting (for SRAM-PUFs). While the average min-entropy (67%) is not as high as some of the PUF-based approaches, it can be potentially increased by adding hardware or algorithm improvements.

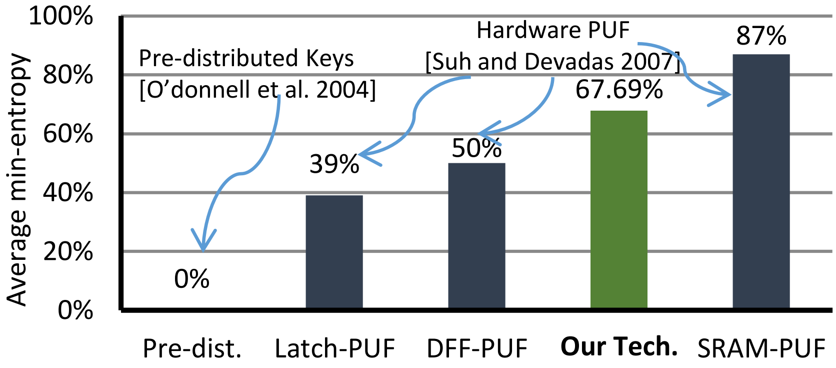


Figure 5.12: Estimated Average Min-Entropy Results Comparison

Performance Overhead Comparison

From the performance point of view, the wireless channel-based key generation algorithm has the advantage of not needing the time-consuming key exchange step of asymmetric and hybrid techniques. Thus, the generation time of the proposed algorithm is compared

[2] According to [51], the average min-entropy increases with the respect to the size of the key.

to the execution time of two of the most popular asymmetric cryptographic algorithms (RSA and ECC [18]) used in hybrid solutions [46]. The comparison is conducted given two different NIST security strength (80 and 112 bits) requirements. The generation times for two different relative velocities (2 mph and 20 mph) from the experiments with the real automotive environment in Section 5.1.3 are used for the comparison. Two cases of RSA and ECC key management were ran on the Raspberry Pi platform, which has a 1.2GHz 64-bit ARMv8 processor: **1)** there is no public-private key generation and pre-installed public keys are simply exchanged, and **2)** new public-private key pairs are generated and exchanged. The first case refers to the possibility that only pre-installed public-private key pairs are used and exchanged but no key generation occurs in RSA/ECC. On the other hand, the second case refers to the possibility that the public-private key pairs are updated with a key generation step and then exchanged in RSA/ECC. The key generation step in asymmetric cryptography is important [2] in order to prevent major security problems such as leakage or eventual reconstruction of the private keys. In fact, in many security protocols involving public-private key pairs for asymmetric encryption, a method for key generation is specified or required.

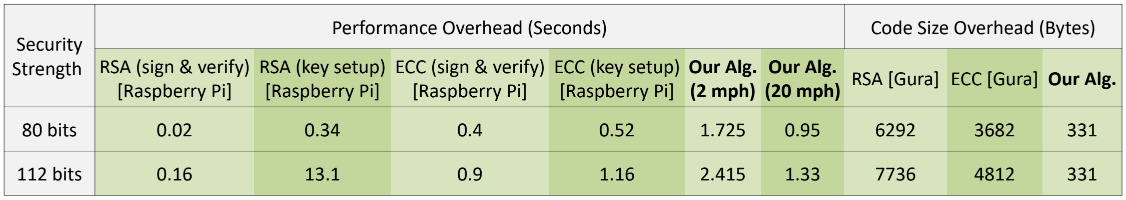
The performance overhead comparison results are provided in Table VI. The first two columns from the left under Performance Overhead correspond to the two cases where no key generation occurs but the key exchange, sign and verify steps do occur. They reveal that the algorithm is much slower than both RSA and ECC given the assumption that a public-private key pair is already established and only key exchange, signing and verifying occur. The results in the two adjacent columns to the right under Performance Overhead correspond to the case where a key generation step occurs before key exchange, signing and verifying in RSA/ECC. These results demonstrate that the approach performs considerably closer to and better than RSA/ECC. Given different scenarios in V2X communication where the communication session may last for several seconds or minutes, the Algorithm 3 will be able to find a proper PSK length (which should be small in most scenarios) so that the security requirements will be met while the key generation time is negligible. The performance of the key generation approach for higher relative speeds mainly depends on how many PSK bits need to be generated and has a linear relationship with security strength.

In summary, although considerably slower than RSA/ECC key management where there is no key generation step, the proposed approach replaces both the public-private key pair generation and exchange steps of asymmetric cryptographic approaches and can run faster than even RSA for certain scenarios. Specifically, in comparison to RSA/ECC, the approach has the following advantages: 1) the advantage of optimizing the key generation time based on the scenario, whereas RSA/ECC will take approximately the same amount of time for each public-private key pair generation step in any type of scenario, and 2) the advantage of a dynamic key generation technique based on physical randomness, whereas RSA/ECC may use a pre-installed and static public/private key pair that is not updated by key generation (or very infrequently). The current results demonstrate that the proposed technique can be a fair alternative method for V2X communication from a performance perspective.

Code Size Overhead Comparison

In order to evaluate the overhead from the memory size point of view, the code size of the algorithm is also compared to the sizes of implemented RSA and ECC key management algorithms. For a fair comparison, the proposed key generation algorithm code was cross-compiled to make it suitable for the same processor, that the RSA and ECC algorithms were ran on, and to get a valid code size. As shown in Table 5.5, the proposed algorithm code is 10 times (10X) smaller than ECC code and is 20 times (20X) smaller than RSA code [18]. Additional code, including that of the key length optimization algorithm and the cryptographic key derivation methods, are also negligible in size and easily programmable onto the constrained devices.

Table 5.5: Performance and Code Size Overhead Comparisons Results



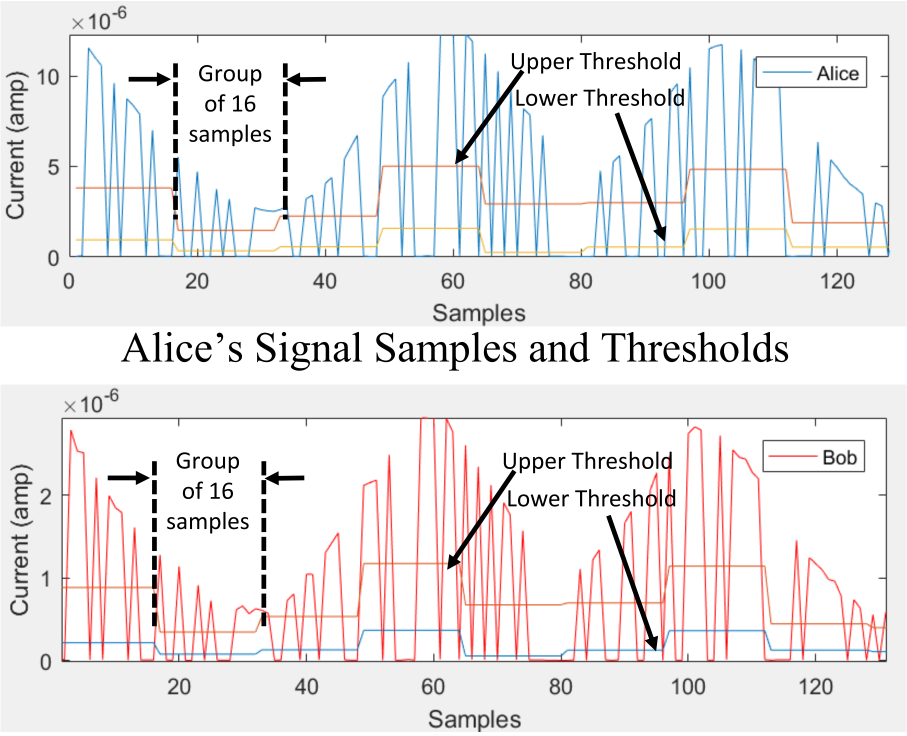
Security Strength	Performance Overhead (Seconds)						Code Size Overhead (Bytes)		
	RSA (sign & verify) [Raspberry Pi]	RSA (key setup) [Raspberry Pi]	ECC (sign & verify) [Raspberry Pi]	ECC (key setup) [Raspberry Pi]	**Our Alg. (2 mph)**	**Our Alg. (20 mph)**	RSA [Gura]	ECC [Gura]	**Our Alg.**
80 bits	0.02	0.34	0.4	0.52	1.725	0.95	6292	3682	331
112 bits	0.16	13.1	0.9	1.16	2.415	1.33	7736	4812	331

5.2 Vehicular Visible Light Communication

5.2.1 Pre-Shared Key Generation

Matlab was used to model the signal propagation between the vehicles, the reciprocity checks, entropy calculation, noise calculations and the key generation algorithm. US 101 (Hollywood Freeway) data from NGSIM was used to extract vehicle position-related information and included it in the mathematical model. For the key generation phase, simulation involves the usage of a 1Kbps probe signal. To minimize the simulation time, a $2^7 - 1$ Pseudo Random Bit Sequence (PRBS) is used as the probe signal and 512 bits to emulate data propagation. When the key generation is done, the vehicles can communicate just as in a conventional VLC link. Generated keys of different lengths with parameters of group size $g = 16$, and threshold constant $\alpha = 0.3$, are shown in Figure 5.13 For the aforementioned settings, after generating 34 keys, 16 of them were mismatching. However, in total, there were only 60 key bits that were mismatching out of 4352 generated key bits. For $g = 16$ and $\alpha = 0.8$, the mismatch rate drops to only one mismatching key out of 20.

For the Alice-Eve channel, with the same PRBS (as Eve is assumed to have all the information about the algorithm), some keys that the adversary Eve generated are shown in Figure 5.14. All 34 keys that Eve produced from the Alice-Eve channel did not match with the 34 keys produced by the Alice-Bob channel. Correlations of the received signals from the channels are also computed.



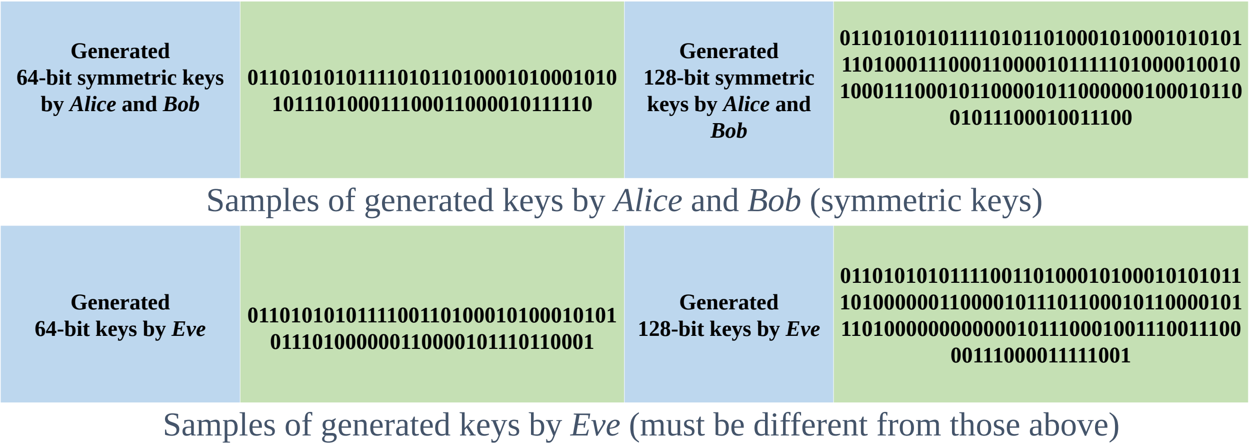
Alice’s Signal Samples and Thresholds

Bob’s Signal Samples and Thresholds

Figure 5.13: Signal Samples and Thresholds for Key Generation

5.2.2 Evaluation

The correlation between Alice’s received signals from Bob and Bob’s received signals from Alice is 0.77 ($Corr_{Alice-Bob,Bob-Alice}$). Correlation between Alice’s received signals from Eve and Eve’s signals from Alice is 0.78 ($Corr_{Alice-Eve,Eve-Alice}$). Correlation between Alice’s received signals from Bob and Eve’s received signals from Alice is 0.32 ($Corr_{Alice-Bob,Alice-Eve}$). The first two values demonstrate the reciprocity property of the channel model while the last value demonstrates that the two channels, Alice-Bob and Alice-Eve, are uncorrelated to each other. Since there was not a sufficient amount of data to use the min-entropy metric (that was used for the RF-based simulations) to evaluate the randomness of the resulting keys, several NIST randomness tests were applied instead [4]. For different key generation settings there are different levels of randomness. Randomness of a key is represented by a set of p-values returned by the NIST tests. For a sequence of bits to be considered truly random, these p-values must be greater than 0.01. 34 keys were generated under the same settings as shown in Figure 5.13 and evaluated. The tests and p-values are: approximate entropy (.148), frequency (.044), block frequency (.044), cumulative sums (fwd=.213, bkwd=.009), runs (.804), FFT (.491), and serial (.499, .425). 31 out of the 34 generated keys passed every single test. Besides the promising results, there is room for future work which may consist of studying parameter optimization and key generation evaluation with larger quantities of data.



Generated 64-bit symmetric keys by *Alice* and *Bob*	**01101010101111010110100010100010101011110100011100011000010111110**	**Generated 128-bit symmetric keys by *Alice* and *Bob***	**011010101011110101101000101000101010111010001110001100001011111010000100101000111000101100001011000000010001011001011100010011100**

Samples of generated keys by *Alice* and *Bob* (symmetric keys)

Generated 64-bit keys by *Eve*	**0110101010111100110100010100010101011101000000110000101110110001**	**Generated 128-bit keys by *Eve***	**01101010101111001101000101000101010111010000001100001011101100010110000010111010000000000000101110001001110011100001110000111111001**

Samples of generated keys by *Eve* (must be different from those above)

Figure 5.14: Examples of Generated Symmetric Keys ($g = 16$, $\alpha = 0.3$)

Chapter 6

Summary and Conclusion

In this thesis, a physical layer symmetric cryptographic key management scheme for vehicular wireless communication is presented. The scheme focuses on exploiting the physical randomness of the vehicular wireless communication channel to efficiently generate strongly random keys for all types of applications. This thesis first develops and presents system models for vehicular radio frequency and visible light wireless communication necessary for key generation. Then, an optimization algorithm is developed to optimize the inputs to the key generation step in terms of length, time and/or energy according to the constraints of different scenarios. Lastly, the proposed physical layer key generation algorithm is developed and presented. The algorithm takes the system models and parameters as inputs to quantize key bits from the signal strength values of the wireless channel. The scheme is a low-cost solution, in terms of performance and code size, for the challenging key exchange problem and confidentiality requirements in vehicular wireless communication applications (particularly safety-critical). As demonstrated by the results, for the radio frequency domain, the proposed algorithm can generate keys with up to 67% average min-entropy. For the visible light domain, the algorithm can generate keys that promisingly pass several NIST randomness tests. In addition, the proposed scheme can be up to 20 times smaller in code size compared to state-of-the-art hybrid cryptographic algorithms such as those in RSA and ECC. In summary, this thesis presents a work on developing a simple yet powerful proof of concept for a practical wireless channel-based symmetric cryptographic key generation technique that can coexist with existing state-of-the-art methods to improve the security and performance of automotive CPS.

Bibliography

[1] B. Azimi-Sadjadi, A. Kiayias, A. Mercado, and B. Yener. Robust key generation from signal envelopes in wireless networks. In *Proceedings of the 14th ACM conference on Computer and communications security*, pages 401–410. ACM, 2007.

[2] E. Barker, W. Barker, W. Burr, W. Polk, and M. Smid. Recommendation for key management-part 1: General (revised. In *NIST special publication*. Citeseer, 2006.

[3] E. Barker and A. Roginsky. Transitions: Recommendation for transitioning the use of cryptographic algorithms and key lengths. *NIST Special Publication*, page 131A, 2011.

[4] L. E. Bassham III, A. L. Rukhin, J. Soto, J. R. Nechvatal, M. E. Smid, E. B. Barker, S. D. Leigh, M. Levenson, M. Vangel, D. L. Banks, et al. Sp 800-22 rev. 1a. a statistical test suite for random and pseudorandom number generators for cryptographic applications. 2010.

[5] P. Belanovic, D. Valerio, A. Paier, T. Zemen, F. Ricciato, and C. Mecklenbrauker. On wireless links for vehicle-to-infrastructure communications. *Vehicular Technology, IEEE Transactions on*, 59(1):269–282, Jan 2010.

[6] G. Blinowski. Security issues in visible light communication systems. *IFAC-PapersOnLine*, 48(4):234–239, 2015.

[7] M. Bloch, J. Barros, M. R. Rodrigues, and S. W. McLaughlin. Wireless information-theoretic security. *IEEE Transactions on Information Theory*, pages 2515–2534, 2008.

[8] K. Bogsjö, K. Podgórski, and I. Rychlik. Models for road surface roughness. *Vehicle System Dynamics*, 50(5):725–747, 2012.

[9] S. Bono, M. Green, A. Stubblefield, A. Juels, A. D. Rubin, and M. Szydlo. Security analysis of a cryptographically-enabled rfid device. In *USENIX Security*, volume 5, pages 1–16, 2005.

[10] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, T. Kohno, et al. Comprehensive experimental analyses of automotive attack surfaces. *USENIX Security Symposium*, 2011.

[11] M.-C. Chuang and J.-F. Lee. Team: Trust-extended authentication mechanism for vehicular ad hoc networks. *IEEE Systems Journal*, 8(3):749–758, 2014.

[12] C. V. S. C. Consortium. *Vehicle Safety Communications Project: Task 3 Final Report: Identify Intelligent Vehicle Safety Applications Enabled by DSRC.* National Highway Traffic Safety Administration, Office of Research and Development, Washington, D.C., 2004.

[13] K. Dar, M. Bakhouya, J. Gaber, M. Wack, and P. Lorenz. Wireless communication technologies for its applications. *Communications Magazine, IEEE*, 48(5):156–162, 2010.

[14] K. Dar, M. Bakhouya, J. Gaber, M. Wack, and P. Lorenz. Wireless communication technologies for its applications [topics in automotive networking]. *Communications Magazine, IEEE*, 48(5):156–162, May 2010.

[15] T. ElBatt, C. Saraydar, M. Ames, and T. Talty. Potential for intra-vehicle wireless automotive sensor networks. In *Sarnoff Symposium, 2006 IEEE*, pages 1–4. IEEE, 2006.

[16] I. ETSI. Intelligent transport systems (its); security; threat, vulnerability and risk analysis (tvra). Technical report, ETSI TR 102 893, European Telecommunications Standards Institute, 2010.

[17] S. Goldwasser and M. Bellare. Lecture notes on cryptography, 2001.

[18] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz. Comparing elliptic curve cryptography and rsa on 8-bit cpus. In *Cryptographic hardware and embedded systems-CHES 2004*, pages 119–132. Springer, 2004.

[19] J. Harding, G. Powell, R. Yoon, J. Fikentscher, C. Doyle, D. Sade, M. Lukuc, J. Simons, and J. Wang. Vehicle-to-vehicle communications: Readiness of v2v technology for application. Technical report, 2014.

[20] D. Hofheinz and E. Kiltz. Secure hybrid encryption from weakened key encapsulation. In *Advances in Cryptology-CRYPTO 2007*, pages 553–571. Springer, 2007.

[21] D. E. Holcomb, W. P. Burleson, and K. Fu. Power-up sram state as an identifying fingerprint and source of true random numbers. *IEEE Transactions on Computers*, 58(9):1198–1210, 2009.

[22] R. M. Ishtiaq Roufa, H. Mustafaa, S. O. Travis Taylora, W. Xua, M. Gruteserb, W. Trappeb, and I. Seskarb. Security and privacy vulnerabilities of in-car wireless networks: A tire pressure monitoring system case study. *19th USENIX Security Symposium, Washington DC*, pages 11–13, 2010.

[23] S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy. On the effectiveness of secret key extraction from wireless signal strength in real environments. In *Proceedings of the 15th annual international conference on Mobile computing and networking*, pages 321–332. ACM, 2009.

[24] H. Krawczyk. Cryptographic extraction and key derivation: The hkdf scheme. In *Annual Cryptology Conference*, pages 631–648. Springer, 2010.

[25] N. Lawson. Highway to hell: Hacking toll systems, 2008.

[26] J. Lee Rodgers and W. A. Nicewander. Thirteen ways to look at the correlation coefficient. *The American Statistician*, 42(1):59–66, 1988.

[27] J. Li, H. Lu, and M. Guizani. Acpn: a novel authentication framework with conditional privacy-preservation and non-repudiation for vanets. *IEEE Transactions on Parallel and Distributed Systems*, 26(4):938–948, 2015.

[28] C.-W. Lin, L. Rao, P. Giusto, J. D'Ambrosio, and A. Sangiovanni-Vincentelli. An efficient wire routing and wire sizing algorithm for weight minimization of automotive systems. *Proceedings of the 51st Annual Design Automation Conference (DAC'14)*, pages 1–6, 2014.

[29] P. Luo, Z. Ghassemlooy, H. Le Minh, E. Bentley, A. Burton, and X. Tang. Fundamental analysis of a car to car visible light communication system. In *Communication Systems, Networks & Digital Signal Processing (CSNDSP), 2014 9th International Symposium on*, pages 1011–1016. IEEE, 2014.

[30] P. Luo, Z. Ghassemlooy, H. Le Minh, E. Bentley, A. Burton, and X. Tang. Performance analysis of a car-to-car visible light communication system. *Applied Optics*, 54(7):1696–1706, 2015.

[31] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik. Radio-telepathy: extracting a secret key from an unauthenticated wireless channel. *Proceedings of the 14th ACM international conference on Mobile computing and networking*, pages 128–139, 2008.

[32] MathWorks. Matlab, simulink. *www.mathwork.com*, 2014.

[33] C. Miller and C. Valasek. A survey of remote automotive attack surfaces. *Black Hat USA*, 2014.

[34] C. Miller and C. Valasek. Remote exploitation of an unaltered passenger vehicle, 2015.

[35] M. A. Moharrum and A. A. Al-Daraiseh. Toward secure vehicular ad-hoc networks: a survey. *IETE Technical Review*, 29(1):80–89, 2012.

[36] A. Mukherjee, S. A. A. Fakoorian, J. Huang, and A. L. Swindlehurst. Principles of physical layer security in multiuser wireless networks: A survey. *CoRR*, abs/1011.3754, 2010.

[37] C. W. O'donnell, G. E. Suh, and S. Devadas. Puf-based random number generation. *In MIT CSAIL CSG Technical Memo*, 2004.

[38] N. Patwari, J. Croft, S. Jana, and S. K. Kasera. High-rate uncorrelated bit extraction for shared secret key generation from channel measurements. *IEEE Transactions on Mobile Computing*, 9(1):17–30, 2010.

[39] N. R. Potlapally, S. Ravi, A. Raghunathan, and N. K. Jha. A study of the energy consumption characteristics of cryptographic algorithms and security protocols. *IEEE Transactions on Mobile Computing*, 5(2):128–143, 2006.

[40] S. N. Premnath, J. Croft, N. Patwari, and S. K. Kasera. Efficient high-rate secret key extraction in wireless sensor networks using collaboration. *ACM Transactions on Sensor Networks (TOSN)*, page 2, 2014.

[41] V. Punzo, M. T. Borzacchiello, and B. Ciuffo. On the assessment of vehicle trajectory data accuracy and application to the next generation simulation (ngsim) program data. *Transportation Research Part C: Emerging Technologies*, 19(6):1243–1262, 2011.

[42] Y. Qian and N. Moayeri. Design of secure and application-oriented vanets. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, pages 2794–2799. IEEE, 2008.

[43] K. Ren, H. Su, and Q. Wang. Secret key generation exploiting channel characteristics in wireless communications. *Wireless Communications, IEEE*, 18(4):6–12, 2011.

[44] M. Rostami, J. B. Wendt, M. Potkonjak, and F. Koushanfar. Quo vadis, puf?: trends and challenges of emerging physical-disorder based security. *Proceedings of the Design, Automation and Test in Europe Conference and Exhibition 2014 (DATE'14)*, page 352, 2014.

[45] B. Schoettle. A market-weighted description of low-beam headlighting patterns in the us: 2004. 2004.

[46] T. Schütze. Automotive security: Cryptography for car2x communication. In *Embedded World Conference*. Citeseer, 2011.

[47] H. Schweppe, Y. Roudier, B. Weyl, L. Apvrille, and D. Scheuermann. Car2x communication: securing the last meter-a cost-effective approach for ensuring trust in car2x applications using in-vehicle symmetric cryptography. In *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, pages 1–5. IEEE, 2011.

[48] M. K. Simon and M.-S. Alouini. Digital communication over fading channels. *John Wiley & Sons*, 2005.

[49] L. Stibor, Y. Zang, and H.-J. Reumerman. Evaluation of communication distance of broadcast messages in a vehicular ad-hoc network using ieee 802.11p. In *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, pages 254–257, March 2007.

[50] G. E. Suh and S. Devadas. Physical unclonable functions for device authentication and secret key generation. *Proceedings of the 44th annual Design Automation Conference (DAC'07)*, pages 9–14, 2007.

[51] R. van den Berg. *Entropy analysis of physical unclonable functions.* PhD thesis, MSc. thesis, Eindhoven University of Technology, 2012.

[52] J. Wan, A. Canedo, A. Faruque, and M. Abdullah. Functional model-based design methodology for automotive cyber-physical systems. *IEEE Systems Journal*, 2014.

[53] J. Wan, A. B. Lopez, and M. A. Al Faruque. Exploiting wireless channel randomness to generate keys for automotive cyber-physical system security. In *Proceedings of the 7th International Conference on Cyber-Physical Systems*, ICCPS '16, pages 13:1–13:10, Piscataway, NJ, USA, 2016. IEEE Press.

[54] J. Wan, A. B. Lopez, and M. A. Al Faruque. Physical layer key generation: Securing wireless communication in automotive cyber-physical systems. *ACM Trans. Cyber-Phys. Syst.*, 2018. To appear in a forthcoming issue.

[55] Q. Wang, H. Su, K. Ren, and K. Kim. Fast and scalable secret key generation exploiting channel phase randomness in wireless networks. In *INFOCOM, 2011 Proceedings IEEE*, pages 1422–1430. IEEE, 2011.

[56] C. Weiß. V2x communication in europe–from research projects towards standardization and field testing of vehicle communication technology. *Computer Networks*, 55(14):3103–3119, 2011.

[57] D. Work, A. Bayen, and Q. Jacobson. Automotive cyber physical systems in the context of human mobility. In *National Workshop on high-confidence automotive cyber-physical systems*, pages 3–4, 2008.

[58] C. Ye, S. Mathur, A. Reznik, Y. Shah, W. Trappe, and N. B. Mandayam. Information-theoretically secret key generation for fading wireless channels. *IEEE Transactions on Information Forensics and Security*, pages 240–254, 2010.

[59] I. U. Zaman, A. B. Lopez, M. A. A. Faruque, and O. Boyraz. A physical layer security key generation technique for inter-vehicular visible light communication. In *Advanced Photonics 2017 (IPR, NOMA, Sensors, Networks, SPPCom, PS)*, page SpTu1F.3. Optical Society of America, 2017.

[60] B. Zan, M. Gruteser, and F. Hu. Key agreement algorithms for vehicular communication networks based on reciprocity and diversity theorems. *IEEE Transactions on Vehicular Technology*, 62(8):4020–4027, 2013.

[61] K. Zeng, D. Wu, A. J. Chan, and P. Mohapatra. Exploiting multiple-antenna diversity for shared secret key generation in wireless networks. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE, 2010.