FedUni ResearchOnline

<https://researchonline.federation.edu.au>

Copyright Notice

This paper is a postprint of a paper submitted to and accepted for publication in [2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks – Supplemental Volume \(DSN-S\)](#). The copy of record is available at the IEEE Xplore Digital Library.

Jolfaei, A., & Kant, K. (2019). Privacy and Security of Connected Vehicles in Intelligent Transportation System. In 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks – Supplemental Volume (DSN-S) (pp. 9–10). IEEE.

<https://doi.org/10.1109/DSN-S.2019.00010>

Copyright © 2019 IEEE

Privacy and Security of Connected Vehicles in Intelligent Transportation System

Alireza Jolfaei

School of Science, Engineering and Information Technology
Federation University Australia
Mt Helen, Victoria 3350, Australia
Email: a.jolfaei@federation.edu.au

Krishna Kant

Department of Computer and Information Sciences
Temple University
Philadelphia, Pennsylvania 19122, USA
Email: kkant@temple.edu

Abstract—The paper considers data security and privacy issues in intelligent transportation systems which involve data streams coming out from individual vehicles to road side units. In this environment, there are issues in regards to the scalability of key management and computation limitations at the edge of the network. To address these issues, we suggest the formation of groups in the vehicular layer, where a group leader is assigned to communicate with group members and the road side unit. We propose a lightweight permutation mechanism for preserving the confidentiality and privacy of sensory data.

Keywords—Intelligent Transportation; Data Security; Permutation; Privacy; Scalability

I. INTRODUCTION

As shown in Fig. 1, the Intelligent Transportation System (ITS) uses a hierarchical architecture, where the bottom layer includes sensors/actuators (IoT devices), such as the mass air flow sensor, the engine speed sensor, oxygen sensor, manifold absolute pressure sensor, and voltage sensor. The second layer includes groups of vehicles that drive as a group together. The vehicles in a group use Vehicle-to-Vehicle (V2V) communications to coordinate their decisions with one another. The third layer includes road zones that can include many groups of vehicles. In this layer, vehicles are subscribed to a Road Side Unit (RSU) through Vehicle-to-Infrastructure (V2I) communications. The RSU monitors road zones and communicates the road rules (speed limit and road toll) and hazards to the vehicles. The fourth layer includes local regions that include a larger area with many zones. Finally, the top layer includes the central traffic control system that collects data from vehicles across all regions, such as an area of a city, and provides longer-term data storage and deep analytics.

II. SECURITY AND PRIVACY ISSUES

There are several security concerns in this environment. Since at least some of the information, for example, driving behavior, is sensitive, the information must be protected during transit between vehicle and RSU and between RSUs. Given the real-time streaming of large amounts of data from vehicles, standard encryption mechanisms, such as AES, may be difficult to implement at the device level. Thus, it is necessary to devise lightweight algorithms for encryption. A related issue is that of remote storage of data taken from vehicles. For this, each vehicle's data needs to be encrypted before storage at

Fig. 1. Hierarchical architecture of ITS.

a remote node, and yet nodes that have access to this data should be able to access it. This again requires a lightweight encryption scheme. Also, managing keys individually for each device could become a burden and a source of weakness. Notice that if each vehicle can subscribe to any RSU, we need enough keys for each pairing of a vehicle and RSU. We also need a group keying mechanism so that the RSUs that have access to the vehicle data can freely share it without repeated encryption/decryption. Each device and node also needs an authentication mechanism to avoid impersonation; however, since authentication is likely done very infrequently, traditional mechanisms may be adequate for it.

The data collected from various sensors are spatially and temporally correlated in nature. For example, individual drivers do not necessarily want to be identified when supplying their data since such data can be misused in numerous ways. For example, if a car provides its ID and speed, it can be easily identified whenever it violates the speed limit. The goal of vehicles is to use the services provided by RSUs while preserving their anonymity as much as possible.

The ETSI TC ITS security standards specify a range of latency requirements for various types of vehicular communication messages [1]. The most stringent latency requirement, applicable to messages conveying imminent safety danger is 20 ms [2], [3]. With a typical 200B message size, this amounts to a security processing rate of more than 10 KB/sec (there are transmit and queuing delays) on both transmit and receive sides. Furthermore, given a typical safety message rate of 10/sec/vehicle, and a communication range of 300 meters, a vehicle may need to verify about 1000 messages/sec. In the worst case, several of these messages may arrive at the

receiving vehicle almost simultaneously, and an even faster processing would be required to meet the 20 ms delay limit. Authentication and encryption at such high rates would require specialized processors and hinder V2V deployment.

III. A POTENTIAL SECURITY APPROACH

In this section, we consider a specific situation where the vehicles can form groups and choose a leader that collects data from other group members and communicates with the RSUs. The group formation is done both to reduce key management overheads and to obfuscate device identities. There are a large variety of effective approaches for group formation as well as group leader selection [4], whereby different approaches typically focus on different performance metrics. For our application, an important performance metric is the size of the group – a group should be large enough to obfuscate device identities well but small enough from the latency and leader bandwidth perspective. The chosen leader should be a device that is less restricted in computing power, storage, energy, and transmission bandwidth. A deployment friendly to our scheme may even choose to deploy devices such that a few of them are powerful enough to serve as leaders, while others are chosen as less powerful as an offsetting mechanism.

We now discuss the obfuscation mechanism. Consider a set of n vehicles that are subscribed to a specific leader. In any time slot t , each device i collects a data chunk, which we assume contains K_i consecutive data stream segments. It sends this data to the leader. After receiving data from all n vehicles, the leader prepares a “matrix” of data segments D_t as follows:

$$D_t = \begin{bmatrix} d_{11} & d_{12} & \dots & d_{1K_1} \\ d_{21} & d_{22} & \dots & d_{2K_2} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \dots & d_{nK_n} \end{bmatrix}, \quad (1)$$

where d_{ij} denotes the j th segment of the chunk sent by the i th vehicle. This matrix is then encrypted and sent to the RSU to which this leader is subscribed.

In [5], we have proposed a lightweight, permutation-only encryption scheme for use in transmitting smart-grid protection messages that are normally not encrypted due to high latency associated with normal encryption mechanisms. In this paper, we propose a simple variant of such an algorithm for encrypting D_t , which we henceforth call as messages.

To permute a given message, we first construct a sequence of pseudo-random numbers by concatenating three pseudo-random sequences generated from a linear congruential pseudo-random number generator, defined by

$$z_{j+1} = (h \cdot z_j + q) \mod 2^{n_0}, \quad (2)$$

where the seed z_0 is arbitrarily chosen, n_0 is the period of the generator, $h \mod 4 = 1$, and $q \mod 2 = 1$. Here the index j corresponds to the j th random number produced. Let $K = \sum_{i=1}^n K_i$. Obviously, $0 \leq j \leq \lceil \frac{K-1}{3} \rceil$, since we want K random numbers from three streams. The pseudo-random sequence is then sorted in a descending order. This provides

a unique index order for relocating each *field* in the message. For example, if z_j ends up in position m after sorting, it means that the j -th field in the message will be relocated to the m -th position. Note that the contents of each field remain unmodified and in the clear.

The recurrence relations can be computed very fast with minimal memory to preserve the state. This allows the simulation of multiple independent streams. The key is randomized with a salt value to avoid using the same permutation mapping repeatedly. Since the security margin of the proposed permutation scheme is $\lceil \log_2(K-1) \rceil$ chosen queries, the key is salted every $\lceil \log_2(K-1) \rceil - 1$ message communications. The salting must be done by both the transmit and receive side so that the two sides remain synchronized. The salting helps to resist certain types of cryptanalysis, such as plaintext attacks and precomputed rainbow table attacks [6]. For example, for $K = 128$, the key is salted every ten transmissions using a counter, which is 32 bits in most microprocessors.

A. Performance Analysis

The computational complexity of the proposed method is the sum of complexities of the pseudo-random number generation and the radix sorting procedure. Our algorithm was tested on a 48-MHz ARM Cortex-M0 microcontroller along with other popular schemes such as AES-128-CCM, AES-128-EAX, AES-128-GCM, Chaskey, MD5, and ChaCha20-Poly1305 on the same platform [7]. The detailed performance is discussed in [5]. It is clear that our method is much faster than all of the competitors and is suitable for real-time use.

IV. CONCLUSIONS

In this paper, we described the security and privacy issues of connected vehicles in ITS and sketched an initial solution to the problem. The security solution uses a lightweight permutation scheme which is suitable for encrypting real-time data streams generated by weak devices. The encryption is done by a leader device which allows device ID obfuscation, easier key management, and allows operation on encrypted data. In the future, we want to consider more elaborate mechanisms and test their scalability.

REFERENCES

- [1] ETSI TS 102 940, “Intelligent transport systems (ITS), security, ITS communications security architecture and security management,” V0.0.13 (2012-03), 2012.
- [2] R. Bertini, “Redefining mobility: Connected/automated vehicles,” in *California Transportation Commission, California Road Charge Pilot Program Technical Advisory Committee (TAC)*, Irvine, CA, 2015.
- [3] R. Williams, C. Cadzow, H. Fisher, and D. Green, “Evaluation of the european cits platform including threat, vulnerability and risk analysis,” in *Austroroads Report*, 2018, pp. 1–126.
- [4] A. A. Abbasi and M. Younis, “A survey on clustering algorithms for wireless sensor networks,” *Computer communications*, vol. 30, no. 14-15, pp. 2826–2841, 2007.
- [5] A. Jolfaei and K. Kant, “A lightweight integrity protection scheme for low latency smart grid applications,” *Computers and Security Journal (COSE)*, Oct 2018.
- [6] P. Oechslin, “Making a faster cryptanalytic time-memory trade-off,” in *Annual International Cryptology Conference*, 2003, pp. 617–630.
- [7] J. Birr-Pixton, “Benchmarking modern authenticated encryption on 1 devices,” <https://jbp.io/2015/06/01/modern-authenticated-encryption-for-1-euro/>, 2015.