On-the-fly Analytics over Encrypted Records in Untrusted V2X Environments

Denis Ulybyshev*, Servio Palacios*, Ganapathy Mani, Aala Oqab Alsalem, Bharat Bhargava
Computer Science Department, CERIAS
Purdue University
West Lafayette, USA
{dulybysh, spalacio, manig, alsalema, bbshail}@purdue.edu

Puneet Goyal
Computer Science and Engineering
Indian Institute of Technology
Ropar, India
puneet@iitrpr.ac.in

***Abstract*— In Vehicle-to-Everything (V2X) networks, vehicles can communicate and exchange data related to road events, such as traffic jams and accidents, road constructions and safety warnings. This capability is used in Intelligent Transportation System applications, aiming to provide road assistance and more safety. However, V2X networks and autonomous vehicles, are a target for attackers. A malicious message sent to a vehicle can put the vehicle in jeopardy. Thus, it is essential to provide data confidentiality and integrity in untrusted V2X communication environments. Also, it is highly desirable for an Intelligent Transportation System to have a capability of building fast-speed data analytics over vehicle records, including encrypted records.**

In this paper, we propose a solution for decentralized data analytics that can be built over encrypted data on local nodes in V2X communication systems. Our solution provides confidentiality and integrity of data, as well as data leakage detection/prevention capabilities for several types of leakages made by insiders to unauthorized parties in V2X networks. Furthermore, we enhance our model through a novel mix of convolutional neural networks and Attribute-Based Encryption (ABE). This improved model diminishes the attack surface of the impersonation and forgery attacks.

I. Introduction

In V2X networks vehicles and roadside objects can communicate and share data with each other. It is critical to provide data confidentiality and integrity in V2X communication systems. To address this, we use a self-protected Vehicle Data Record (VDR) which incorporates encrypted datasets in the form of key-value pairs, access control and metadata policies and policy enforcement engine. VDR uses the concept of Active Bundle [1], [2], [3]. VDR provides role-based and attribute-based access control [4]. VDR is used for data exchange in vehicular networks without necessity of having central authority to enforce access control policies [5]. In addition, VDR provides capabilities of detecting and preventing data leakages, that could be made by authorized parties to unauthorized ones in V2X networks [6]. In this paper, we propose a method to perform decentralized on-the-fly data analysis over encrypted data records. Additionally, this paper provides alternative solutions to alleviate the effect of impersonation and forgery attacks. The former implies that vehicles disguise as different entities to alter the behavior of cars or network [7]. Forgery attack means that the vehicles send incorrect or false messages (e.g., emergency messages, warnings, etc.) that can alter the normal behavior of the network [7]. We provide an enhancement to our security model that can diminish the possibility of those attacks. Our solution comprises a novel mix of previous work on convolutional neural networks [8] and Attribute-Based Encryption (ABE) [9], [10]. The idea is to capture unique attributes of the vehicle and, based on them, derive a key which is used to encrypt and decrypt a dataset, incorporated in VDR, used for data analysis. We utilize the Make, Model, and Color Recognition (MMCR) [8] to detect the vehicle's attributes and construct an access tree that serves as a guarantee of the existence of that node in the network. This paper has two main contributions:

1) Methodology to build decentralized data analytics over encrypted data on local nodes in V2X communication system. Our solution provides confidentiality and integrity of data, as well as data leakage detection/prevention capabilities for several types of leakages made by insiders to unauthorized parties in V2X networks [6]. Our secure data transfer method works for both centralized and decentralized peer-to-peer network architectures, which is essential for V2V communication system. Our approach supports role-based and attribute-based access control, as well as large subset of SQL queries over encrypted data [11].
2) A novel key derivation technique utilizing convolutional neural networks (i.e., for make, model, and color recognition) and Attribute-Based Encryption (ABE) to diminish the attack surface for impersonation and forgery attacks.

The rest of the paper is organized as follows: section II presents related work. Section III presents the core design of our system. Section IV evaluates performance of the system. Section V concludes the paper.

II. Related Work

A. Attribute-Based Encryption

Goyal et al. [9] introduced Key-Policy Attribute-Based Encryption (KP-ABE) in which the key defines the access

***Both authors contributed equally and are considered to be co-first authors.**

structure (i.e., keys determine the ciphertext that they are allowed to decrypt). Also, a set of descriptive attributes serve as labels in the ciphertext. In the Ciphertext-Policy ABE (CP-ABE) the access policy is included in the ciphertext [10]. It has been shown that CP-ABE is computationally expensive [12]. CP-ABE requires an access tree based on the attributes of the data.

B. Vehicle Image Classification

Liu and Wang utilize large datasets to train and test classifiers [13]. Dehghan et al. introduce Sighthound [8] a vehicle make, model, and color recognition system. Sighthound relies on a deep convolutional neural network. In this paper, we utilize Sighthound JavaScript API [14] to retrieve results from the classifier.

C. Secure Data Exchange in V2X networks

European standards (ETSI) and US standards (WAVE) for Intelligent Transport System (ITS) have privacy requirements for vehicles data. ETSI require anonymity, pseudonymity, unlinkability and unobservability of vehicles data [15]. Ranchal et al [16] proposed an EPICS framework to protect data privacy throughout the service interaction lifecycle. This solution relies on Active Bundle [2], [1], [3] which transfers data together with the access control policies, specified by data owner, and with an execution monitor that controls data accesses. In this paper, we extended this approach with capabilities of performing on-the-fly data analytics over encrypted data stored in Active Bundle. We also came up with a novel mix of previous work on convolutional neural networks [8] and Attribute-Based Encryption (ABE) [9], [10] to derive encryption key.

D. Performing on-the-fly local data analysis

Due to the on-the-move nature of autonomous vehicles in V2X environment, decentralized computations are the important components of the core autonomous vehicle-to-vehicle (V2V) design. Several methods have been proposed in literature to perform decentralized data analysis. A decentralized data neural network has been proposed in [17]. Instead of performing analytics after transferring the data, this method only transfers gradients calculated through backpropagation. This eliminates data transfer from individual entities to a centralized location but performs machine learning analytics through gradients, which also enables privacy-preserving data analytics. Similarly, authors in [18] introduced a decentralized data analysis framework that is assisted by low-cost hardware such as MEMS. The data acquisition and processing are performed at the local MEMS sensor nodes. Results alone are transmitted and used for decision processes. The framework avoids gathering data in a centralized location for data analysis.

III. Core Design

A. Vehicle Data Record

In order to provide data confidentiality and integrity, we use a self-protected Vehicle Data Record (VDR) which incorporates encrypted datasets in the form of key-value pairs, access control and metadata policies and policy enforcement engine. Here is the exampe of key-value pair:
vdr.vehicleOwnerName : Enc(John Doe)
Each dataset is encrypted with a separate AES symmetric key, which is generated on-the-fly based on VDR execution flow. VDR provides privacy protection for vehicle's and vehicle owner's data. An example of a VDR is given in Table 1. VDR relies on the Active Bundle concept [1], [2], [3] and extends it with multiple capabilities, such as:

1) fast on-the-fly data analytics over encrypted data;
2) encrypted search over encrypted data, supporting large subset of SQL queries [11];
3) preventing and detecting data leakages, made by insiders to unauthorized entities [6];
4) attribute-based access control which considers client's authentication method browser cryptographic capabilities [4].

VDR supports authorized on-the-fly data updates.

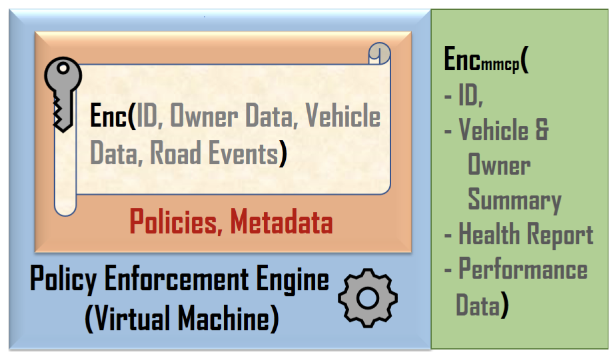

Fig. 1. Vehicle Record.

Communication workflow for services, representing V2X objects and requesting dataset from VDR, is as follows. Firstly, when client requests a particular VDR dataset, e.g. vehicle owner's driving license number, the client's identity is verified. Client presents to VDR its X.509 certificate signed by a trusted Certificate Authority (CA). If authentication passes, client's attributes are evaluated and enforced by the policy enforcement engine, incorporated into VDR. If this evaluation successfully passes, then evaluation of applicable access control policies starts. Based on that, decryption keys are derived to decrypt those datasets for which the client

TABLE I
Vehicle Data Record

PrimKey	Owner Info	Vehicle Info	Road Event
ID	Name	License Plate	Accident
	Home Address	VIN	Obstacle
	Drivers License	Health Report	Traffic Jam
	Phone	Engine parameters	Road Work
	Email	Fluid Level	Weather Alert
		Mileage	
		Tire Pressure	
		Performance Report	
		Average Speed	
		Trip Mileage	

is authorized. Details of communication procedure between web service and Active Bundle are covered in [2]. Demo video, illustrating this concept extended with data leakage capabilities for our implemented prototype is available [19]. "Summary" field of VDR is special since it does not require evaluation of access control policies. It is encrypted with MMCP key, which is derived based on captured attributes of a vehicle, including make, model, color and license plate. Convolutional neural networks technique is utilized. Details are given in section III C below. The service who has MMCP-key can decrypt Summary without going through policy evaluation. It allows to perform fast on-the-fly data analytics. Summary field contains vehicle ID, license plate number, health and performance data, including average speed, trip duration, fuel consumption, engine temperature, etc. Other data, necessary for Intelligent Transportation System, might be included in Summary as well. More sensitive data such as owner's home address are not included in Summary and clients are required to go trough full policy evaluation check to access these sensitive data.

B. System Architecture

In our architecture, there are vehicles, roadside units which transfer data to/from cloud provider, speed cameras and high-resolution toll gate cameras. VDR-based solution supports both centralized and decentralized peer-to-peer network architectures. In our design, V2X communication objects are represented as web services, e.g. Driver, Law Enforcement, Insurance Service, Corporate Brand Vehicle, etc. Client can be a vehicle or the computer which represents Intelligent Transportation System or Law Enforcement. Data exchange scenario for V2X network is shown on Fig. 2 [1].

At step 1, speed camera captures vehicle speed and license plate and sends them together with the speed limit at step 2 to the cloud provider. At step 3, once vehicle reaches the toll gate, the high-resolution camera captures vehicles make, model, color, license plate number and sends these attributes at step 4 to the cloud provider, where they are used to derive a unique encryption MMCP key at step 5. Details of key derivation are given in section III C below. At step 6, this encryption key is sent to the vehicle, along with previously captured at steps 1, 2 pairs of (speed, speed limit). At step 7, all the vehicles attributes are bonded together and Vehicle Record is created, having them in encrypted form [2]. Vehicle Record is created locally at the vehicle, not at the cloud provider, to guarantee protection against malicious or curious cloud administrators. Vehicle Record is used for secure decentralized inter-vehicle data communication (IVC)

[1]* Clip-art taken from:
https://openclipart.org/detail/291705/no-passing-road,
http://www.clker.com/clipart-soft-green-car-1.html,https://clipartuse.com/car-picture-clipart-45635,
https://www.pinterest.com/pin/406027722629862023/,
https://www.clker.com/clipart-basestation.html,
http://clipground.com/image-post/25938-gate-ahead-clipart-4.png.html,
https://commons.wikimedia.org/wiki/File:Database.svg,
https://play.google.com/store/apps/details?id=com.DC.speed.camera,
http://worldartsme.com/brown-car-clipart.html,
https://www.iconfinder.com/icons/305819/

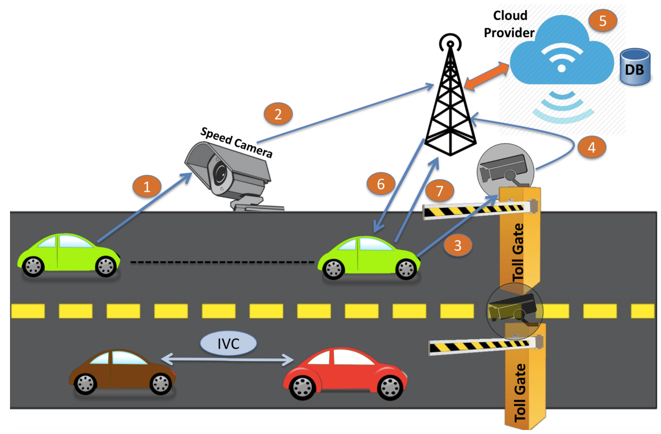

Fig. 2. *V2X Communication Network

in wireless V2X networks. Another application of VDR is local data analysis. Authorized vehicles with MMCP key, e.g. insurance service car, can query nearby vehicles, insured in a given company, for summary statistics and build local analytics, e.g. on vehicles performance and health. Insurance service car can also build statistics on traffic jams in a certain area in certain hours and cluster vehicles based on their state, retrieved from the license plate. 'Average speed' field of VDR can be used for traffic jam analysis. Such capability of on-the-fly fast data analytics can be useful for Intelligent Transportation System in order to provide better road assistance for the drivers.

Assumptions:

- Hardware platform and OS that run VDR are trusted.
- Https protocol is used for communications between all the web services.

C. Key Derivation

1) Bilinear Maps: Since we use **KP-ABE**, the Bilinear Maps are a building block of our construction. Bilinear Maps include cyclic groups of prime order q. Moreover, a bilinear map is an injective function with the properties of *bilinearity, non-degeneracy, and computability* [9], [10]

2) Key-Policy Attribute-Based Encryption: Ciphertexts in **KP-ABE** are associated with a set of attributes. **KP-ABE** needs an access tree that defines the access policy. Also, the key generation depends on the access tree. The **KP-ABE** includes four algorithms:

- The **setup** algorithm sets the attributes to be considered an outputs a public key PK and a Master Key MK.
- The **encryption** algorithm takes the input m (i.e., a message), the public key PK, and a set of attributes γ and produces a ciphertext CT.
- The **key generation** algorithm uses the master key MK, an access tree τ, and the public key PK to produce a secret key SK so that SK can decrypt CT iff τ matches γ.
- The **decryption** algorithm utilizes the secret key SK, the public key PK, and the ciphertext CT and decrypts CT iff τ matches γ, otherwise returns $\perp$.

3) Automated Highway Toll Systems: Automated highway toll systems utilize sensors and high-resolution cameras that scan vehicle's shape, the number of axles, make, color, model, and license plate [20]. Sensors in the pavement trigger signal to the overhead camera to capture the vehicle's color photo (i.e., these photos help to identify the owner and ensure the correct billing for the toll zone transaction). When the camera fails to capture and recognize the owner and proper billing; some alternatives include a computerized image of the shape and size of the car. We rely on Electronic Toll Systems and toll cameras (sometimes called tool booth cameras with high resolution) to recognize the make, model, color, and license plate. Then, we feed the MMCR system (via RESTful requests to the cloud [14]). This framework has an accuracy of 93.6% on the Stanford car dataset [21]. Since we utilize current infrastructure with high-resolution toll booth cameras, we expect similar accuracy.

Our work is inspired by [12]. Our access control relies on a tree-based access structure embedded in the key (**KP-ABE**). Our nodes/vehicles are described using unique attributes attached to it (e.g., make, model, color, license plate.) We can model multiple use cases (e.g., when the recognition algorithm catches some attributes of the node, but not all) that will imply different reputation level in the system.

The access tree τ depicted in Fig. 3 shows the *attributes* as the *leaves* and *threshold-gates* as *non-leaf* nodes. As mentioned in [12], **OR** and **AND** gates can be represented using ***1-out-of-n*** and ***2-out-of-2*** threshold gates respectively. Therefore, we can express a rich set of rules.

4) Simplified Protocol: Our simplified protocol behaves in the following way. We utilize toll stations (e.g., T_k) since those incorporate high-resolution cameras that can give a better performance for the MMCR [2] algorithm[3].

- Using MMCR algorithm, we build a data structure (i.e., a **JSON** file JS_i) with the set of attributes of the vehicle.
- In the setup phase, the certifying authority (e.g., toll station T_i) generates the master key MK and the public key PK.
- The certifying authority (T_i) creates an access tree τ_i using the attributes of step 1 and adding a random nonce N_i (128-bit) (Fig. 3) that serves as unique id and **session id**. T_i generates the secret key SK_i over τ_i.
- T_i encrypts the secret key SK_i and the random nonce N_i with T_i's public key TPK_i (we call this ciphertext CT_i).
- T_i encrypts the *random session key* with the key policy attribute-based encryption.
- The certifying authority (T_j) decrypts the content of CT_i and obtains the random nonce N_i and the secret key SK_i.
- T_j retrieves a set of attributes (JS_j) utilizing the MMCR algorithm.
- T_j decrypts the random session key utilizing SK_i and the access tree τ_j . Then T_j verifies the correct value and elevates the node to the **Authenticated** state.
- The vehicle receives *symmetric keys* (derived from the session key) to communicate securely.

[2]MMCR refers to Make, Model, and Color Recognition Algorithm.

[3]According to [14] to obtain good results, the vehicles in an image should have at least 200 pixels wide.

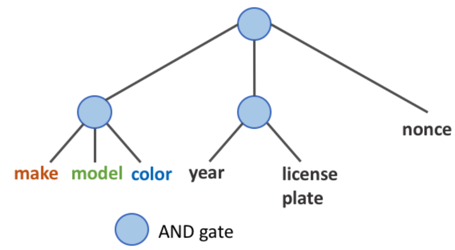

Fig. 3. ABE Access Tree.

D. On-the-fly Machine Learning Analytics

On-the-fly data analytics such as aggregate analytics about the local vehicular speeds and traffic information such as number of tolls or road blocks can aid the decision making process of autonomous vehicles. For example, assessing the relative speed of the surrounding vehicles, the autonomous vehicles can adjust its speed to maximize the efficiency in reaching the destination. Number of cars in each lane can also provide an insight into changing lanes to keep up the speed or accelerate or reduce for safety. Data can be perturbed to preserve the privacy of each vehicle.

Let us consider a use case of CAR_A requesting speed from the neighboring vehicles that are on the move. CAR_A can request the data from other vehicles in the following way: CAR_A sends its perturbed speed quantity $CAR_{A_{speed}}$ + R_{noise} to its neighboring cars.

$$CAR_{A_{speed}} + R_{noise} = CAR_{A_{speed}} + CAR_{B_{speed}} + CAR_{C_{speed}} + = Total_{speed}$$

$$AverageSpeed = \frac{Total_{speed} - R_{noise}}{TotalNumberofCars}$$

The perturbation keeps the data anonymized hence preserving the privacy of individual vehicle. Aggregate analytics such as this are localized in which the local area can be specified by the administrator and the car requests data from the cars that are present in that local area. This reduces computation, communication, and storage overhead of the autonomous vehicle. Even insurance companies and car companies can utilize such analytics to enhance their services. For example, insurance company can calculate the accident rates in a particular area with speeds and the insured vehicles, and alert the policy holder to monitor the autonomous car for safe driving.

Unsupervised learning methods can aid in effective knowledge discovery and cognitive autonomy of the autonomous vehicles. In order to classify streaming data, we need efficient and fault-tolerant system. The scheme should also carry a low computational overhead. Based on clustering through error-correcting codes [22], vehicle records categories (or) features can be classified into 23 bit vectors. Each feature is classified with 23 one-bit attributes (0 or 1 i.e. characteristics of a feature is present or absent in the incoming data). This increases the clustering size and scales the analysis for large number of records. Unlike the traditional clustering mechanism, error-correcting code clustering can be completed in O(N) time.

IV. Evaluation

Section A evaluates round-trip time (RTT) for inter-vehicle communication made in the form of VDR. RTT is measured between the moments when service, representing a vehicle, issues data request to another vehicle and retrieved data are received at the recipient's side. RTT includes authentication, authorization, key derivation and data disclosure phases. ApacheBench, ver.2.3, utility is used on client side to send series of data requests. In section A we also compare latency of data request sent to VDR with just decryption of the same dataset without involving evaluation of access control policies and client's attributes. In section B we evaluate the performance of MMCR algorithm for detecting vehicle attributes, including make, model and color. Time required to transfer the captured attributes over the network is also evaluated.

A. Inter-vehicle communication evaluation

In this experiment, we firstly evaluate performance overhead imposed by VDR compared to just decrypting the same dataset. In both cases, client requests the same dataset of 617 bytes, stored in encrypted form. In the first case, encrypted dataset is just decrypted, using AES algorithm. In the second case, request for the same dataset needs to go through evaluation of access control policies and client's attributes, as well as leakage detection check. Our selected VDR incorporates four access control policies and uses AES algorithm to encrypt and decrypt the stored data. As it can be seen from Fig. 4, VDR imposes 102% performance overhead compared to just decryption of encrypted dataset. In both cases, we used Raspberry Pi Model B as a hardware platform, with ARMv7 Processor rev 4 @1.2GHz, RAM 1GB, Raspbian GNU/Linux 9.1 (stretch) operating system. We assume that Raspberry Pi board, which has small credit-card size and moderate power consumption, represents vehicle's communication hardware.

Next, we evaluate inter-vehicle data exchange made in the form of VDR and compare performance of VDR with Vehicle Data Record Baseline (VDRB). VDRB, in contrast with VDR, does not support data leakage detection, fast on-the-fly data analytics and data dissemination, based on cryptographic capabilities of client's browser. In addition, VDRB is not tamper-resistant. We varied the number of access control policies, included in VDRB. Experiments cover 1, 2, 4, 8 and 16 access control policies. VDR, used in our prototype, contains four access control policies. Thus, conclusion about VDR performance overhead is made based on comparing RTT for VDRB(4) and VDR. Each vehicle is represented by Raspberry Pi. The first vehicle hosts a VDR and listens to the opened port 5555. The service, representing

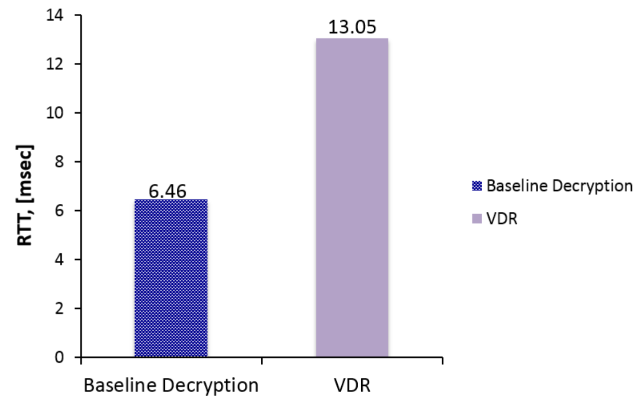

Fig. 4. Inter-vehicle Communication Round Trip Time (initial request).

the second vehicle, sends request for 617 bytes of data to the first vehicle over wireless TCP/IP network. Measured RTT includes network delays.

Vehicle 1 (Raspberry PI 3 Model B)
Hardware: ARMv7 Processor rev 4 @1.2GHz, RAM 1GB
OS: Raspbian GNU/Linux 9.1 (stretch), IP: 128.10.120.158
Vehicle 2 (Raspberry PI 3 Model B)
Hardware: ARMv7 Processor rev 4 @1.2GHz, RAM 1GB
OS: Raspbian GNU/Linux 9.1 (stretch), IP: 128.10.120.240

We run 50 data requests in a row, using ApacheBench, ver.2.3, utility. As it can be seen from Fig. 5, VDR adds 127% performance overhead, compared to baseline VDRB(4).

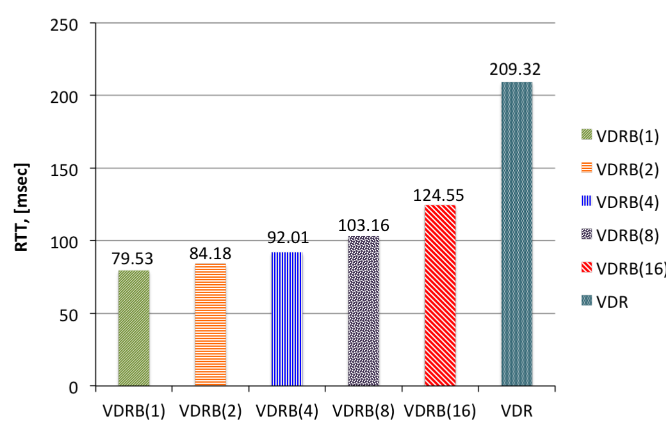

Fig. 5. Inter-vehicle Communication Round Trip Time.

B. Vehicle recognition and transfer time evaluation

In this experiment, we measure the total running time for the MMCR algorithm. We set our environment with a modest Internet Connection (i.e., 12 Mbps download, 1 Mbps Upload, the latency is around 53 ms.). Also we craft multiple (i.e., a hundred) requests to the cloud provider (i.e., Sighthound RESTful API [14]). The goal is to simulate a real scenario in which we query the cloud (the cloud will contain our trained model) and return the results to the certifying authority (e.g., toll station). In Fig. 6 we observe that the total time per request is around one second. We receive a JSON file with the inferred attributes of the car.

This experiment exposes the feasibility of our proposed solution.

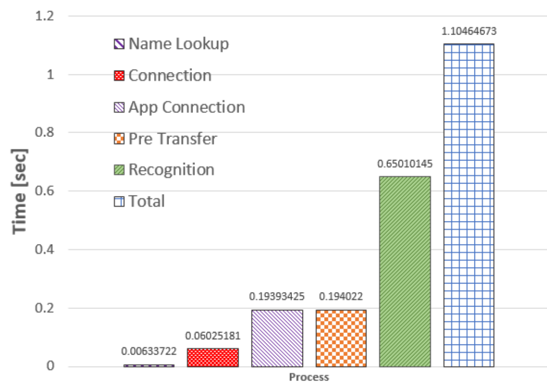

Fig. 6. Vehicle recognition and transfer time.

V. Conclusion

We presented a privacy-preserving decentralized data exchange model for V2X communication systems, which, in addition to data confidentiality and integrity, provides capability to perform local on-the-fly data analytics. Our solution supports role-based and attribute-based access control, as well as detection of data leakages, made by insiders to unauthorized entities in V2X networks. Transaction latency for data message sent in the form of VDR between two vehicles is 209 msec. Support of attribute-based access control, tamper resistance, data leakage detection capability and fast local on-the-fly analytics capability add 127% performance overhead. VDR concept, that involves evaluation of access control policies (in our use case, four policies) and client attributes, as well as data leakage check, imposes 102% performance overhead compared to just decryption of encrypted dataset without evaluating access control policies and client attributes, and without data leakage check. Our approach demonstrates that security features for inter-vehicle data communication can be implemented without overshadowing the safety features. Extensive experimental results and more use cases will be included in our future publications. We enhanced our model through a novel mix of vehicle recognition algorithms and attribute-based encryption. Hence, we provide an alternative solution for alleviating the impersonation and forgery attacks. Our experimental results demonstrate the feasibility of our idea.

Acknowledgment

This publication was made possible by NPRP grant # [7-1113-1-199] from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors. The authors would like to thank Dr. Leszek Lilien and Dr. Lotfi Ben Othmane for their help and valuable feedback.

References

[1] L. B. Othmane, *"Active bundles for protecting confidentiality of sensitive data throughout their lifecycle", Ph.D dissertation.* Western Michigan University, 2010.

[2] R. Ranchal, *"Cross-domain data dissemination and policy enforcement", Ph.D dissertation.* Purdue University, 2015.

[3] L. Lilien and B. Bhargava, "A scheme for privacy-preserving data dissemination," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 36, no. 3, pp. 503–506, 2006.

[4] D. Ulybyshev, B. Bhargava, M. Villarreal-Vasquez, A. O. Alsalem, D. Steiner, L. Li, J. Kobes, H. Halpin, and R. Ranchal, "Privacy-preserving data dissemination in untrusted cloud," in *Cloud Computing (CLOUD), 2017 IEEE 10th International Conference on.* IEEE, 2017, pp. 770–773.

[5] C. Qu, D. A. Ulybyshev, B. K. Bhargava, R. Ranchal, and L. T. Lilien, "Secure dissemination of video data in vehicle-to-vehicle systems," in *Reliable Distributed Systems Workshop (SRDSW), 2015 IEEE 34th Symposium on.* IEEE, 2015, pp. 47–51.

[6] D. Ulybyshev, A. Alsalem, and B. Bhargava, "Secure data exchange and data leakage detection in untrusted cloud," in *ICACCT, 2018. Accepted, in-press.*

[7] B. Mokhtar and M. Azab, "Survey on Security Issues in Vehicular Ad Hoc Networks," *Alexandria Engineering Journal*, vol. 54, no. 4, pp. 1115–1126, 2015.

[8] A. Dehghan, S. Z. Masood, G. Shu, and E. G. Ortiz, "View Independent Vehicle Make, Model and Color Recognition Using Convolutional Neural Network," pp. 1–7, 2017.

[9] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based Encryption for Fine-grained Access Control of Encrypted Data," *Proceedings of the 13th ACM Conference on Computer and Communications Security*, pp. 89–98, 2006.

[10] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," *Proceedings - IEEE Symposium on Security and Privacy*, pp. 321–334, 2007.

[11] D. Ulybyshev, A. Alsalem, B. Bhargava, S. Savvides, G. Mani, and L. Ben-Othmane, "Secure data communication in autonomous v2x systems," in *IEEE ICIOT, 2018. Accepted, in-press.*

[12] V. Kumar and S. Madria, "Distributed Attribute Based Access Control of Aggregated Data in Sensor Clouds," *2015 IEEE 34th Symposium on Reliable Distributed Systems (SRDS)*, pp. 218–227, 2015.

[13] D. Liu and Y. Wang, "Monza: Image Classification of Vehicle Make and Model Using Convolutional Neural Networks and Transfer Learning," 2016.

[14] "Sighthound." [Online]. Available: https://www.sighthound.com/products/cloud

[15] "Standard: ETSI - TS 102 941. Intelligent Transport Systems (ITS); Security; Trust and Privacy Management." [Online]. Available: https://standards.globalspec.com/std/1530232/etsi-ts-102-941

[16] R. Ranchal, B. Bhargava, P. Angin, and L. B. Othmane, "Epics: A framework for enforcing security policies in composite web services," *IEEE Transactions on Services Computing*, 2018.

[17] N. Lewis, S. Plis, and V. Calhoun, "Cooperative learning: Decentralized data neural network," in *Neural Networks (IJCNN), 2017 International Joint Conference on.* IEEE, 2017, pp. 324–331.

[18] E. Uhlmann, A. Laghmouchi, C. Geisert, and E. Hohwieler, "Decentralized data analytics for maintenance in industrie 4.0," *Procedia Manufacturing*, vol. 11, pp. 1120–1126, 2017.

[19] "Secure Data Dissemination prototype demo video." [Online]. Available: https://www.dropbox.com/s/4wg3vuv52j4s16v/NGCRC-2017-Bhargava-Demo1.wmv?dl=0

[20] "The Phantom Tollbooth." [Online]. Available: https://www.raytheon.com/news/feature/electronic_tolling

[21] "Car Dataset." [Online]. Available: https://ai.stanford.edu/~jkrause/cars/car_dataset.html

[22] S. Sarangi and S. Banerjee, "Efficient hardware implementation of encoder and decoder for golay code," *IEEE transactions on very large scale integration (VLSI) systems*, vol. 23, no. 9, pp. 1965–1968, 2015.