A Privacy-Preserving Solution for Intelligent Transportation Systems: Private Driver DNA

Gianpiero Costantino, Marco De Vincenzi, Fabio Martinelli, and Ilaria Matteucci

***Abstract*— The rising connection of vehicles with the road infrastructure enables the creation of data-driven applications to offer drivers customized services. At the same time, these opportunities require innovative solutions to protect the drivers' privacy in a complex environment like an Intelligent Transportation System (ITS). This need is even more relevant when data are used to retrieve personal behaviors or attitudes. In our work, we propose a privacy-preserving solution, called *Private Driver DNA*, which designs a possible architecture, allowing drivers of an ITS to receive customized services. The proposed solution is based on the concept of Driver DNA as characterization of driver's driving style. To assure privacy, we perform the operations directly on sanitized data, using the *Order Revealing Encryption* (ORE) method. Besides, the proposed solution is integrated with ITS architecture defined in the European project *E-Corridor*. The result is an effective privacy-preserving architecture for ITS to offer customized products, which can be used to address drivers' behaviors, for example, to environmental-friendly attitudes or a more safe driving style. We test Private Driver DNA using a synthetic dataset generated with the vehicle simulator CARLA. We compare ORE with another encryption method like *Homomorphic Encryption* (HE) and some other privacy-preserving schemas. Besides, we quantify privacy gain and data loss utility after the data sanitization process.**

***Index Terms*— ITS, privacy, driver DNA, order revealing encryption, homomorphic encryption.**

I. Introduction

IN RECENT years, Intelligent Transport Systems (ITS) are experiencing increasing digitalization with the Internet of Things (IoT) and the application of new technologies like next 5G New Radio (5G NR). Consequently, the growth of connected devices and the expanding networks are increasing the demand for cyber protection and privacy [1]. In the last years, both aerial and ground transport system communications have received greater attention and have been largely studied. For example, the new 5G NR with its lightweight and energy saving could be applied in Mobile Ad-hoc Networking (MANET), in Vehicle Ad-hoc Networking (VANET), and also in Flying Ad-hoc Networking (FANET) [2], showing the strong correlation among the different transport networks.

Manuscript received 3 August 2021; revised 4 February 2022, 20 April 2022, and 30 August 2022; accepted 13 October 2022. Date of publication 10 November 2022; date of current version 26 January 2023. The Associate Editor for this article was H. H. Guest. *(Corresponding author: Gianpiero Costantino.)*

Gianpiero Costantino, Marco De Vincenzi, and Ilaria Matteucci are with the Institute of Informatics and Telematics, 56124 Pisa, Italy (e-mail: gianpiero.costantino@iit.cnr.it).

Fabio Martinelli is with the Department of CNR, Institute of Informatics and Telematics, Consiglio Nazionale delle Ricerche, 56124 Pisa, Italy.

Digital Object Identifier 10.1109/TITS.2022.3217358

In this work, we focus on ground vehicles within an ITS and we follow the ITS definition contained in the directive of the European Union 2010/40/EU, where ITS are "*advanced applications, which […] aim to provide innovative services relating to different modes of transport and traffic management and enable various users to be better informed and make safer, more coordinated and 'smarter' use of transport networks*" [3]. Following this definition, we design a privacy-preserving solution to provide innovative personalized services, based on a rewarding process, as it is described in the European project *E-Corridor*. In particular, *E-Corridor* aims to develop a technological framework to unleash the power of information sharing, coupled with edge-based collaborative analytics for cyber protection [4]. The framework allows *prosumers* (producers/consumers) to express their consent to share data with the infrastructure and receive customized prices, based on their driving style, which can be described as a personal driver's usual method of driving, defined using several vehicle's parameters in a long-term driving [5]. In our study, the driving style is defined with a metric, called Driver DNA. As defined in [6] and described in §III, Driver DNA is composed of four parameters (breaking, turning, speeding, revolutions per minute), that can be retrieved from the in-vehicle network: each parameter, combined with the others, can represent a particular driving attitude. For instance, the parameter speeding, which is calculated using also the weather conditions, can be used to determine the safeness of a driving style in a specific road segment.

Our infrastructure can identify a user's driving style and compare it with the other drivers in a privacy-preserving environment. The U.S. National Institute of Standards and Technologies (NIST) [7] states that the "*privacy is the right of a party to maintain control over and confidentiality of information about itself*". Hence, the privacy of a user's driving style needs to be guaranteed with a full control over data. In our E-Corridor infrastructure, the control is defined in the Data Sharing Agreement (DSA) that a driver signs with the infrastructure, so a person can exercise the rights over personal data.

In this paper, we leverage the ITS architecture employed in the E-Corridor project and propose a privacy-preserving solution, called Private Driver DNA. It aims to define an effective process that carmakers and infrastructure authorities can implement in a real road environment. The main question to which our paper tries to answer is: *in a ground ITS infrastructure, how could we provide customized privacy-preserving*

services to drivers? Our solution enables answering the rising demand for driver personalized services, providing a complete architecture for driving style aware services. In particular, the privacy-preserving solutions are guaranteed by the execution of operations directly on sanitized data, using the Order Revealing Encryption (ORE) method.

To evaluate the feasibility, we test Private Driver DNA on synthetic data generated with the open-source autonomous driving simulator *CARLA* [8]. We compare the ORE method with Homomorphic Encryption (HE), as another possible encryption method and we compare our schema with other current schemas for VANETs. Finally, we quantify the privacy gain and the data loss utility of the proposed privacy-preserving solutions.

A. Motivations and Contributions

In an ITS, the demand of custom-fit services is quickly increasing, and, today, users can choose several solutions to receive personalized services. For instance, Connected Cars [9] is a mobile app that enables the communication between workshops and car owners to provide customized services, based on users' data. DriveQuant [10] is another mobile app that analysis the driving style and helps drivers to adopt safer driving behaviors and reduce fuel consumption. Another vehicle data-centered application is Zenroad [11], which is a free tracking and safe-driving app to collect location, driving style, driving behaviors, and driving patterns to monitor and analyze the vehicle status. However, this kind of solutions can suffer several security and privacy attacks. For instance, in 2017, Kaspersky's Mikhail Kuzin and Victor Chebyshev [12] tested several apps that receive data from the vehicle and they showed that an attacker can gain access to the app and that it is possible to get the GPS coordinates, trace the route, steal vehicle's data or performing malicious actions like unlocking the doors. In this scenario data privacy has become one of the main topics in automotive researches like [13], [14], [15], [16],and [6]. Besides, the increasing connection of vehicles with the road infrastructure has required a deep study on the availability of privacy-preserving services. Moreover, our paper is motivated by the E-Corridor project that aims to provide a privacy-aware ITS framework to achieve confidential, distributed and edge-enabled private services in a multi-modal transport systems. With all these motivations, we aim to show the feasibility of a full privacy-preserving process to offer drivers customized services within an ITS.

To the best of our knowledge within the automotive, our work represents the first privacy-preserving solutions that allows drivers to receive services based on their driving style in a full ITS environment like the E-Corridor project provides. The Private Driver DNA, applied in a rewarding system and based on their driving style, can preserve user's data privacy because data are encrypted before sending them out of the vehicle, and the computations, thanks to the ORE method, are performed over encrypted data. As a practical contribution, with the rewarding process, our solution may address drivers' behaviors, for example, to environmental-friendly attitudes or a more safe driving style, with benefits for all drivers and the community.

Our solution inherits from [6] the metric to define the driving style starting from only four parameters, the Driver DNA, but we supersede this work, using the metric in a privacy-preserving process with encryption methods for computations, and providing a feasible service solution. In our work, we decide to combine an ITS infrastructure inherited from the E-Corridor project and an encryption method to provide a complete privacy-preserving ITS infrastructure, where prosumers can share their data without privacy loss. As a contribution test, we make a comparison between ORE and HE method to define the best suitable solution for our infrastructure. We test the proposed privacy-preserving solution with CARLA simulator. Then, we compute the *privacy gain*, which is quantified as "the degree of uncertainly that the original data can be inferred from the sanitized one", using the Shannon entropy [17]. Finally, as a subsequent step, we evaluate the *data loss utility*, defined as "the amount of information that we lost with privacy mechanism" [18].

B. Structure of the Paper

§ II discusses the related work and § III reports about the cornerstones of our privacy-preserving infrastructure. § IV is divided into four parts where § IV-A describes the attacker model, § IV-B defines the environment where the attacker can operate, § IV-C describes the possible threats for our infrastructure and, finally, § IV-D provides a risk analysis with the countermeasures. In § V, we describe our infrastructure, the interactions among the different components and the Driver Secure Identifier (DSI) code. In § VI, we test our solution on a simulated vehicle dataset, generated with software CARLA. Firstly, we apply ORE (§ VI-C), while, secondly, we apply HE (§ VI-D) to evaluate the most suitable encryption method for our infrastructure. § VII-A and § VII-B compute how many pieces of information have been disclosed and if data, after applying the protection methods, can still be useful for analytics. § VIII is a comparison between ORE and HE methods, showing that ORE could be the best solution for our infrastructure. § IX compares ORE with different schemas for VANETs to define the advantages and limitations of ORE. § X draws the conclusion and possible future research lines.

II. Related work

We can identify two main categories of related work: the privacy-preserving schemas for ITS and the metrics to evaluate the driving styles for providing personalized services. In the following subsections, we describe each category with the references to the works that have been considered and compared with our study.

A. Privacy-Preserving Schemas for ITS

In the last years, several studies have been performed on secure authentication for air and ground ITS. In particular, three surveys, [19], [20], and [21] focus on privacy-preserving authentication for ITS and provide possible classifications. To order the related work, we adopt the first survey classification schema, where Jan et al. identify different privacy-preserving method categories. We choose this schema

because, in addition, to the categorization, it provides also a reproducible comparison schema where we can add our work to be compared with the other existing schemas as reported in §IX.

1) Pseudonym Based Authentication: in this category pseudonyms are used as privacy-preserving solutions for VANETs. For example, Li et al. [22] defines vehicle pseudonyms using differential privacy methods. However, concerning this category, our solution does not use a pseudonym to identify a vehicle, but it uses multiple data to identify not only the vehicle but also the driver, providing more information than a simple pseudonym.

2) Blockchain-Based Authentication: another significant and rising schema category is the Blockchain-Based Authentication, where blockchain technology is used to authenticate the nodes and for the operations of the network. For example, in [23] and [24] the authors propose frameworks for secure and safe integration of Unmanned Aircraft Systems (UAS) into complex networks, using blockchain and simulating realistic wireless communication scenarios. Blockchain is also proposed as a possible solution for communications in VANETs, where Feng et al. [25] describes an efficient and scalable blockchain-assisted privacy-preserving authentication system for VANETs. In [26], the authors propose another framework to provide authentication automatically in VANETs, but preserving privacy. Besides, as our work, this last schema does not require any online registration center (except for system initialization and vehicle registration). All previously cited solutions use blockchain technologies and decentralized systems, while our schema is a hybrid decentralized method, which follows the pilots needs of the *E-Corridor* project. In this way, the architecture can have a central server, but also a multi-node structure to authenticate, store and analyze data. Moreover, our solution is designed to reduce the number of involved actors to be effective and authenticate the nodes. On the contrary, an approach with blockchain may have required more effort on that aspect.

3) Group Signature Based Authentication: the Group Signature Based Authentication category provides schema with multiple participants to authenticate and/or analyze data. Ren et al. [27] define a crowdsourcing scheme based on reinforcement learning to assure data quality and privacy. Li et al. [28] propose a secure and privacy-preserving navigation schema, using vehicular crowdsourcing based on fog-based VANETs to fulfill the security and privacy requirements of authentication. Concerning this category, our schema does not need a group signature to authenticate or recruit enough participants to perform data-based sensing tasks, but it is based on a few nodes with a more efficient solution for our context because it can work also with only a vehicle and a near RSU.

4) Cryptography Mechanisms: this category contains the schemas which use only cryptographic solutions to assure privacy. Kong et al. [29] studied HE as a privacy-preserving mechanism. The authors design a complete verifiable querying scheme in vehicular fog data dissemination, applying the HE Pallier cryptosystem with a Trusted Authority (TA), which generates the keys for data encryption. In our solution, we do not rely on a TA and, after the tests, we decide to use an ORE schema to directly encrypt data into the vehicle. Qinglong et al. [30] propose a complete privacy-preserving schema with a TA to register vehicles on the network, without using any pseudonym. In our HE test, to avoid the presence of a TA like in [29], keys are generated directly into the vehicle, using one of the most mature and well-defined HE schemes like CKKS, which supports approximate computations of real numbers [31]. Even if in our work we study two different cryptographic mechanisms, ORE and HE, to find the most efficient for our context, our schema provides an hybrid solution for the authentication like with the driver secure identifier which means that our scheme can fit better in the following category.

5) Identity Based Authentication: our work can be classified in the Identity Based Authentication category, where the identifier is derived from driver and vehicle information. Zhang et al. [32] use a privacy-preserving authentication framework that combines 5G and edge computing technology. This work mainly focuses on authentication protocol, while our work focuses more on the definition of a privacy-preserving framework, using ad hoc protocols and enabling users to receive customized services. Wei et al. [33] propose a privacy-preserving protocol for VANETs, using outsourcing computing and identity-based signature. The authors use also an homomorphic mapping to achieve the security, while in our work we prefer not using homomorphic solutions which can bring high computational costs.

B. Driving Style Metrics

Concerning the driving style evaluation, Jurecki et al. [34] define the driving profile starting from two vehicle parameters like frontal and lateral acceleration. Our driving profile is based on a well-defined metric like the Driver DNA with four parameters and it is defined into a complete ITS architecture, while Jurecki's work defines only the metric definition. Lefevre et al. [35] study the driving style to predict the future driver inputs and provide personalized driving assistance, while, in our work, we do not use forecast systems and we provide customized services.

Several types of research describe how to find a driver identifier, starting from several personal or vehicle data like in [36], [37] and [38], but the identifier is not enclosed in infrastructure for customized services, as we design in our schema. Different researches define only privacy-preserving communication protocols for vehicular ad hoc networks (VANETs) like in [39] and [40]. This scenario confirms that the study of data privacy to receive personalized services in ITS is a rising topic, but, often, research focuses only on a single component or scenario. In particular, to the best of our knowledge, only a few studies [35] define a complete privacy-preserving architecture for customized services like our infrastructure.

III. Background

A. Driver DNA

As a driving style-aware solution, we apply Driver DNA [6], which defines a driver driving style profile. Generically

speaking, the DNA (Deoxyribonucleic Acid) is used as the chemical name for the molecule that carries genetic instructions in all living things [41]. It is composed of two strands that wind around one another and that are built with only four bases (adenine, cytosine, guanine, and thymine). In [6], the authors borrow the idea of this 4-bases structure to create a DNA composed of four parameters retrieved from the vehicle. A vehicle with its thousands of sensors generates gigabytes of data per hour [42], so, during the driving of the vehicle, the prosumer generates the necessary data to calculate the DNA. In particular, Driver DNA as defined in [43] is composed of four parameters that can be retrieved from the Controller Area Network (CAN) of the vehicle and each parameter represents a particular driving attitude. As defined in [6], the first parameter is "*breaking*", which can be used to quantify driver aggressiveness and comfort driving. The second parameter is *turning*, measured from the variation of the lateral acceleration. This parameter quantifies the aggressiveness in turning and the use of the steering wheel [6]. The third parameter is *speeding*, which defines the safety of a driving style, combined also with the weather conditions. The fourth parameter is *revolutions per minute (RPM)*, measured with an integer or float number, defining the fuel consumption [6].

In our solution for each of the four parameters, all $\overline{x}_i$ values, retrieved in a specific time or distance window j, are then averaged together to obtain a unique score $\overline{m}_j$ that can define a driver attitude. Then, as suggested in [6], the unique score $\overline{m}_j$ should be compared with other drivers' values to have a comparison baseline. For these reasons, as we define in our infrastructure, it should be created a database containing all the $\overline{m}_j$ values of drivers, divided, as suggested in [43], in quantiles q_1,..., q_6, assigning a score $k \in \{1, 2, 3, 4, 5\}$ to each $\overline{m}_\mathrm{j}$ such that $q_k < \overline{m}_j \leq q_{k+1}$ (with q_1 as smallest limit in the distribution and a score of 1 if $\overline{m}_\mathrm{j} = q_1$). With this process, we obtain an instant integer score k for each specific time window or road, that can represent the driver's attitude with respect to other drivers.

B. Order Revealing Encryption

As a privacy-preserving solution, in addition to the encryption methods for the secure communication channels, we decided to apply the Order Revealing Encryption (ORE) schema, which allows us to define to which quantile the value k, retrieved from driver DNA, belongs.

In our case, for a specific time or distance window j, we only need to perform comparisons between the mean value of each DNA parameter $\overline{m}_j$ and the quantile limits to find the specific driver rank k. For these comparison operations, we decided to apply ORE following the Chenette schema [44].

An ORE method is an encryption schema, where there is a function to directly compare ciphertexts [44]. An ORE schema, as defined in [44], is a tuple of algorithms $\prod =$ (ORE.Setup, ORE.Encrypt, ORE.Compare) defined over a well-ordered domain D with the following properties:

1) ORE.Setup $(1^\lambda) \rightarrow$ sk. On input a security parameter λ, the setup algorithm ORE.Setup outputs a secret key "sk" for the encryption algorithm.
2) ORE.Encrypt (sk, m) $\rightarrow$ ct. With input secret key "sk" and a plaintext message m $\in D$, the encrypt algorithm ORE.Encrypt outputs a ciphertext c. In particular, let $b_i \ldots b_n$ the binary representation of m and let sk = k. For each $i \in [n]$, the encryption algorithm computes:
$$u_\mathrm{i} = F(k, (i, b_1 b_2 \ldots b_{\mathrm{i}\text{-}1} || 0^{n-1})) + b_\mathrm{i} \ (mod M) \quad (1)$$
It outputs the tuple $(u_1, u_2, \ldots, u_n)$.
3) ORE.Compare $(c_1, c_2) \rightarrow$ b. On input two ciphertexts c_1, c_2, the compare algorithm ORE.Compare outputs a bit b $\in$ 0, 1 as the result of order. Firstly, the compare algorithm parses:
$$c_1 = (u_1, u_2, \ldots, u_n)$$
$$c_2 = (u_1^1, u_2^1, \ldots, u_n^1) \quad (2)$$
where $u_1, \ldots, u_n, u_1^1, \ldots, u_n^1 \in \mathbb{Z}_\mathrm{M}$. Let i be the smallest index where $u_i \neq u_i^1$. If no such index exists, output 0. If such index exists, output 1 if $u_i^1 = u_i$+1 (mod M), and 0 otherwise [44].

IV. Security Assessment

In this section, we introduce and evaluate the considered ITS infrastructure from the security aspects defining the constraints for our analysis. In particular, we present the attacker, the environment where the attacker operates, the threat model and, finally, we perform a risk analysis.

A. Attacker Model

In our scenario, we assume an honest but curious attacker (HBC) to retrieve data and information. The HBC is frequently used in the analysis of privacy properties and, starting from the definition in [45], a HBC can be defined as a "*legitimate participant in a communication protocol who will not deviate from the defined protocol but will attempt to learn all possible information from legitimately received messages*". Following this definition, a HBC can be considered passive because the attacker can not take any action other than trying to learn private information by the observation of protocol execution or using already granted privileges.

B. Attacker Environment

Our architecture, reported in Fig. 1, can be divided into two main environments: the *in-vehicle environment*, represented by the prosumer-vehicle (PV) and the *out-of-vehicle environment*, represented by the edge node (EN), service provider (SP), and E-Corridor storage (ES).

In our scenario, we assume the driver and the vehicle as trusted entities. The involved in-vehicle components are the driver E-Corridor app, the vehicle infotainment system, the electronic control units (ECUs), the vehicle trusted platform units (TPMs), and the related communication channels. We suppose a possible real situation, where these components are not compromised, because there was the application of common security rules like the usage of firewalls, TMP, or safe firmware that can guarantee the not-compromised property. ECUs, and also the other components, can suffer physical and

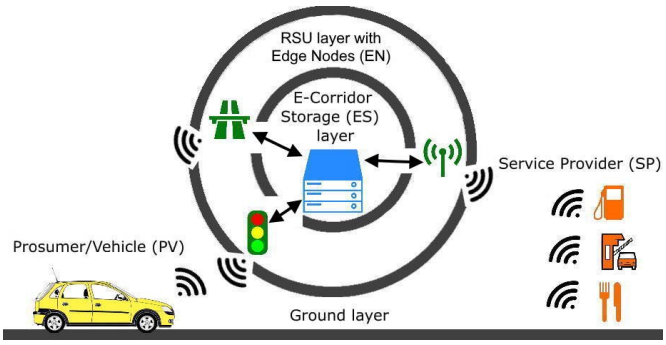


Fig. 1. Privacy-preserving mixed distributed architecture for driver DNA sharing to receive customized services.

local attacks like the injection of malicious frames, but the study of these possible complex attacks is out of the scope of this research and we suppose a not-compromised in-vehicle environment. The second out-of-vehicle environment is composed of every element outside the vehicle. We consider this environment untrusted and unsafe. In Section IV-C, we present the threats that an attacker can perform to compromise the privacy-preserving property.

Besides, in our infrastructure, we will use a third-party application like the app of the mobile Driving Licence (mDL) to retrieve some personal data. This app has been considered like a TA, in fact, the app contains a document worldwide issued by a governmental authority. For these reasons, in our risk analysis for the app mDL we did not propose any countermeasure for the attacks, because we suppose to be a secure environment, where the TA has taken adequate countermeasures.

C. Threats Analysis

In the out-of-vehicle environment, we can identify the following threats for each element of our infrastructure:

- Impersonation:
  - *Service Provider (SP)*: The attacker behaves as Service Provider who aims to steal users' information to profile them for commercial purposes. The attacker has the right to access the database.
  - *Edge Node (EN)*: The attacker, impersonated, for example, by an unfaithful technician, has the right to access the device and eavesdrop on the messages.
  - *E-Corridor Storage (ES)*: The attacker, impersonated, for example, by an unfaithful technician, has the right to access the storage and read data.
- Man-in-the Middle:
  - *Communications channels among PV, EN, ES, SP*: Man-in-the-Middle (MITM) attack. The attacker intercepts communications between two or more actors.

D. Risk Analysis

The previously defined threats can have different occurrence probabilities. In particular, we consider a possible situation in which a SP who tries to retrieve users' data for commercial purposes. To avoid this risk, the SP receives only the DSI without any personal information, so the SP can perform only general statistics, combining the users' data, without being able to associate the DSI with a single real user. Another threat is represented by the MITM attack. To mitigate the risk, messages exchanged within the infrastructure will use symmetric or public-key encryption.

In this analysis, we also consider threats with a minor occurrence probability. In particular, considering an HBC attacker, the only threat, which can affect the EN or the ES, can be someone who has the access privileges like an unfaithful technician. To mitigate this risk, the DSI is represented by a hash digest, without any personal information. Thus, even if an attacker reads it, she will not discover anything more than an alphanumeric code. Besides, as further privacy protection mechanism, the Driver DNA data received by the EN, are encrypted and the decryption is not feasible in the EN because there are no stored passwords. In the ES, Driver DNA data are stored in an encryption format and the E-Corridor passwords are securely stored as hash digests and following the common storing rules for passwords.

V. Private Driver DNA

The core activity of our privacy-preserving architecture is the computation of a *Private Driver DNA* as an enhanced version of Driver DNA with privacy to receive customized services. The environment is represented in Fig. 1 where are defined the four main actors of our architecture: the prosumer with the vehicle (PV), the edge node (EN), the service provider (SP), and the E-Corridor storage (ES).

The PV belongs to the ground layer and generates Driver DNA data. PV is identifiable by E-Corridor infrastructure with its own encrypted Driver Secure Identifier (DSI), a specific deterministic code for identification, retrieved by driver and vehicle data (§ V-B). The EN belongs to the Road Side Unit (RSU) layer and is a generic RSU, which can communicate with the other actors and contain computation elements as described in § V-A. The SP belongs to the ground layer and could be a filling station, a motorway exit, a restaurant, a market, or any service provider registered in the E-Corridor infrastructure. The ES belongs to a cloud layer and it contains all the Driver DNA data history related to a specific DSI. Data in this storage should be stored in an encrypted form. The encryption process has to be performed before the storage by the driver's vehicle with a key that only the driver knows. In this way, the only actor that can decrypt the stored values is the driver with its secret key.

In this architecture, we can provide customized services, based on the vehicle's data, while preserving the driver's privacy. In Fig. 2, we show the prosumer/vehicle's activities and processes to generate data for Private Driver DNA and to receive a service. The start of our workflow is the driving of a vehicle with the consequent data generation. In Fig. 2, we can identify three main processes. The first is the *collection* process, where the four vehicle data to compute Driver DNA (breaking, turning, RPM, and speed) are collected and saved in a vehicle's buffer memory. During this process, data are temporary stored in a buffer memory until data size reach a certain limit, which may depend on storage limitations og the

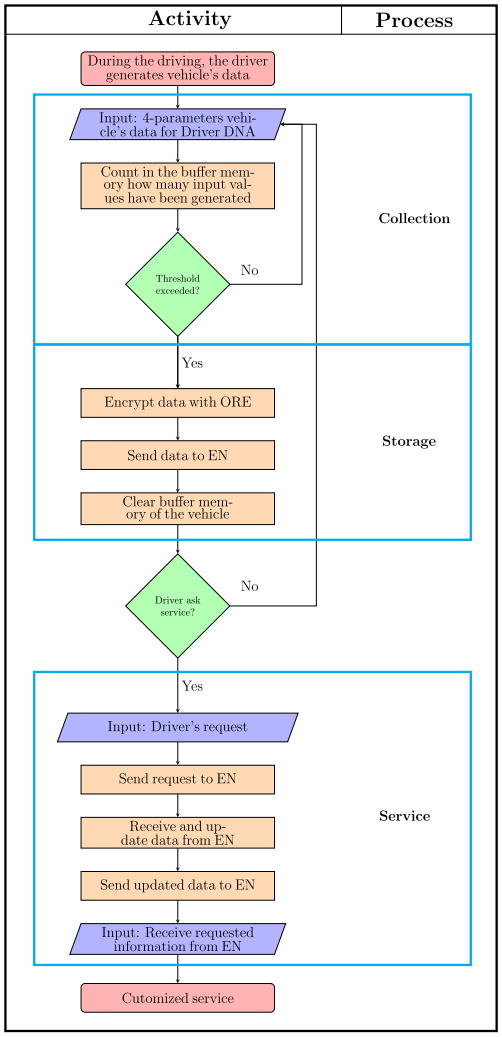


Fig. 2. Process workflow of vehicle's activities and processes to receive a service. To notice that the input "Driver's request" could arrive at any moment of the process, but, for designing reasons, has been inserted after the unload of the vehicle's buffer memory.

device used to collect the data, e.g., the Head Unit. When this limit is reached, the data are sent out of the vehicle and the second process, namely *storage*, starts. In this phase, the data are encrypted with ORE and sent to the EN, which will send data to the ES. Then, the vehicle's buffer memory is emptied to receive new data. The third process starts when the driver asks for a *service*. In Fig. 2, the second decision diamond, for design reasons, is represented after the storage process. This service process sends a request to the EN, which will trigger the process described in Fig. 3. From a prosumer/vehicle perspective, the vehicle receives from EN its historical encrypted data, decrypts the data, uploads the most recent data stored into the memory buffer, and sends back to EN the encrypted data. Using information received from ES, EN computes the best result for the driver's request and send it to the vehicle. The end of the flowchart is the customized service as requested by the driver.

An example of the complete privacy-preserving process is represented in Fig. 3. The UML sequence diagram shows a prosumer which requires to E-Corridor infrastructure a customized prices of fuel among the different nearest filling

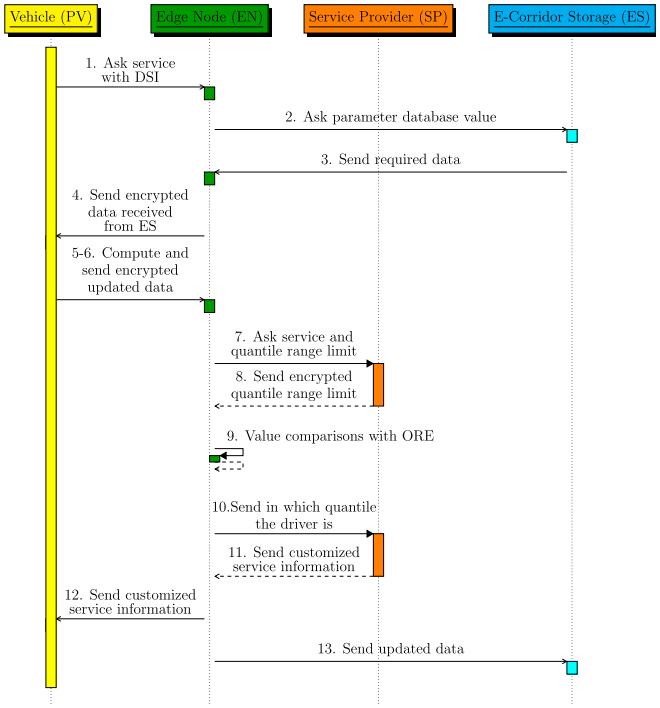


Fig. 3. UML sequence diagram of a request of customized price from a prosumer, driving the vehicle, to a service provider like a charging station. The reported colors are aligned with the colors of Fig. 1.

stations. In this case, the SP is identifiable with the filling station and the following process should be applied for each station involved. The used Driver DNA parameter is *RPM*.

The process interactions (Fig. 3) are the following:

1) PV asks EN a customized price of fuel, sending EN also the DSI.
2) EN asks ES the RPM database of the specific driver, sending ES also the related DSI.
3) ES sends EN the encrypted requested driver's data.
4) EN sends the PV the encrypted data received from ES.
5) PV decrypts the data using the driver secret key and performs, for example, an incremental mean, to update the received data with the most recent RPM value.
6) PV sends EN the encrypted data related to RPM. The encryption method should be previously defined by E-Corridor infrastructure.
7) EN asks SP the quantile range limit related to RPM of the other drivers.
8) SP sends the quantile limit range sanitized with the same encryption standard as driver RPM data, previously defined by E-Corridor infrastructure.
9) Using ORE, EN computes the comparison between the RPM average received from PV and the quantile limits received from SP.
10) EN sends SP the integer score k as defined in §III, giving SP the information to which quartile the driver's RPM value belongs.
11) SP sends EN the plaintext of the customized price using a secure communication channel.

12) EN sends PV the plaintext of the customized price using a secure communication channel.
13) EN sends ES the sanitized updated data received from PV and used for the comparison.

As stated before, this process could be performed for each of the filling stations in the area covered by the RSU. In this way, the prosumer can receive different customized fuel prices and decide which price is more convenient. The difference among the service provider prices could be derived from different quantile limits, which could be based on the different SP policies or different policies of rewarding customers.

Regarding privacy, in our case, using the ORE encryption, we assume just to receive from the vehicle a mean $\overline{m}_j$ in a specific time window or for a defined road distance *j*, computed by the vehicle and associated with a secure driver identifier (DSI) number as defined in § V-B. The only constraint of this process, to apply the ORE method, is that PV and SP use the same sanitization method, which should be previously defined by E-Corridor infrastructure.

A. ITS E-Corridor

As stated in [4], the E-Corridor framework architecture consists of a set of services that can be used in the multi-modal transport scenarios to reach the project targets. In our process, we use the mixed distributed model architecture [4]. In Fig. 1 we report our infrastructure, where data are shared and services are requested by the prosumer. In the E-Corridor defined architecture there are five different subsystems, namely Information Sharing Infrastructure (ISI), Information Analytics Infrastructure (IAI), Data Sharing Agreement (DSA) Lifecycle Infrastructure (DLI), Common Security Infrastructure (CSI), and Advanced Security Infrastructure (ASI). Each subsystem performs a particular task, like data sharing, data analysis, sharing rules definitions, or defining security services [46]. In particular, each subsystem is defined as:

1) ISI subsystem is *"the process of sharing information in a secure way ensuring data isolation and applying privacy-preserving methods on dataset produced sanitized data"* [47] in accordance with a DSA.
2) IAI subsystem can *"receives processed information from ISI and extracts intelligence through data mining and information analysis technique"* [47].
3) DLI manages all life of a DSA, to define rules about information exchange among the different actors of the infrastructure.
4) CSI provides security services like handling cryptographic keys generation and secure storage.
5) ASI manages privileges of network resources and actors to access prosumer information in the infrastructure.

All these subsystems can be found in each actor of the model and, in particular, our architecture is based on ISI, where data are shared among different edges, and on IAI, where data are processed and analyzed. All these operations have to be performed in compliance with each DSA accepted by the prosumers and using different security and privacy-preserving techniques. In particular, the DSA is the document that informs users about the usage of their data and the users' rights. In our architecture, data are not associated with users' personal information but only with the privacy-preserving code DSI, which is the only element to identify the user in the architecture. Besides, data sent out of the vehicle are encrypted and can be used for statistical purposes only in this encrypted form and only associated with DSI. To perform statistics on encrypted data, HE techniques, which can preserve also privacy [29], can be applied, but as we show in § VIII, the results we obtain in our model were less efficient than applying the ORE method.

In our solution, starting from the left side of Fig. 1, we can find the prosumer with the vehicle, that contains all the five described subsystems, in fact, it has to execute sharing and analytic operations. The central actor of our infrastructure is the edge node that can contain and perform the five subsystems described before. The service provider and the E-Corridor storage do not have a local IAI, in fact, they do not perform any analytics on the data, but they only store and share data with the ISI subsystem.

B. Driver Secure Identifier

To design a complete privacy-preserving architecture, it is necessary to describe how a driver and the related vehicle are identified in the network. Hence, we associate the driver and the vehicle in our infrastructure in a deterministic way, enabling the network to identify the driver unequivocally on every possible vehicle she is driving without any possible data leakage. We create a Driver Secure Identifier (DSI), a unique deterministic number that identifies a specific driver in a specific vehicle and assures three different properties: integrity, authenticity, and confidentiality of personal data. We follow the NIST definition of these properties [7]. Besides, we add some additional requirements that we consider to design our DSI:

1) A driver can drive different vehicles.
2) A vehicle can be driven by different drivers (e.g. car sharing).
3) The network (e.g. a service provider) has to be able to distinguish from the DSI the driver and the vehicle that are on the network, so the DSI has to report information on the driver, but also on the vehicle.
4) No use of biometric information of the driver like a fingerprint, iris, face, palm vein, or voice recognition, to show the possibility to create a unique DSI without these sensible data.
5) We suppose that the vehicle environment, composed by a smartphone of the user, infotainment system and the vehicle is not compromised.

After the constraints definition, we design the strategy to create the DSI. To assure adequate protection during the DSI creation, we implement a two-factor authentication process (2FA) [48]. In particular, we use the *ownership factor*, something the user has, such as cards, smartphones, or other tokens, and the *knowledge factor*, something the user knows, such as a password or generically a secret [48].

In our work, as an ownership factor, we decide to use the information that can be retrieved from the Mobile Driving

Licence (mDL), which is the digital representation on a device like a smartphone of the information contained in a physical driver's license. In the next years, the mDL could become a standard to drive a vehicle, because it provides information about the driver's license and, in this way, it enables VANETs to verify the right of the driver to drive, for example verifying the age. Besides, mDL is already tested in several USA states [49] and, to define a unique standard, the ISO/IEC FDIS 18013-5 is under development, being part also of the UN Sustainable Development Goals for 2030 [50].

Thus, we can suppose that a driver can pair with the vehicle the mDL contained in a specific wallet or app in her smartphone. In this way, several data flows from the smartphone to the vehicle and, to create the DSI, we suppose that the infotainment system or in general the vehicle can retrieve the driving license number and the date of birth (DoB) of the driver: the two pieces of information are used to verify the authenticity of the license and the age of the driver. With these details, we retrieve the ownership factor (smartphone and user information with mDL). These two pieces of information should be acceptable from the vehicle only if retrieved from mDL otherwise are not acceptable for DSI generation.

The second factor, the knowledge, can be retrieved from the E-Corridor Password (ePSW) that the user has to insert in the E-Corridor app saved in the infotainment system before starting the drive. The ePSW has to be saved during the first registration of the user on the app E-Corridor as any other app. The ePSW should respect the ideal actual requirements for a safe password like that it must be at least 11 characters long and it must contain at least one upper case, one lower case, one number and one special keyboard character like states in [51]. It could be also used a pattern lock design password to speed up driver password typing, but this solution should be deeper studied with a specific focus on password security. In this paper, we consider that it is sufficient to know that an adequate password, created following for example [51], can be our random and knowledge factor for DSI.

After having retrieved the mDL number, driver DoB, and ePSW, we need a deterministic algorithm that can assure integrity, authenticity and is almost impossible to be reversible for confidentiality property. We choose a keyed hash algorithm like Hash-based Message Authentication Code (HMAC) with SHA-512 like defined in [52]. In our specific case, the hashed part is composed by the concatenation of mDL and DoB numbers, composing several variable lengths depending on the length of mDL number. In the worst case for security, we can assume that the shortest mDL has seven numeric digits in some of US states [53], while the length of DoB should be fixed eight variable digits in the format dd/mm/yyyy. So, for example, the shortest number in the USA to hash could have fifteen digits. We apply as hash algorithm SHA512, which generates a 64-bites digest [54], [55] and it provides strong resistance to collision and preimage attacks and is assumed to remain secure in the dawning era of quantum computers [56]. Then, as in the definition of HMAC, it is necessary to introduce a random secret like the ePSW that makes our DSI almost unique and with randomness making it safe against a brute-force attack [52]. In our case, the ePSW

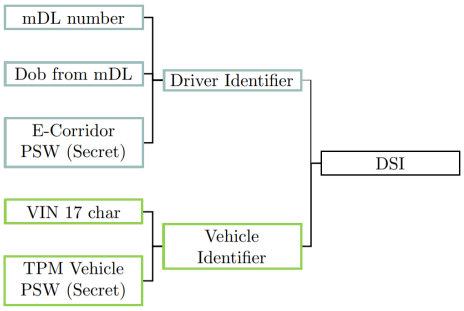


Fig. 4. DSI composition with the application of HMAC SHA512 on driver and vehicle information and then concatenate the obtained hash digests.

with its composition as defined before has a low probability to be discovered and SHA512 can have the characteristics to be a *"minimally reasonable hash functions"*. Hence, a hash function like HMAC with SHA-512 allows us to have a deterministic identifier that, at our best knowledge, is irreversible protecting the personal information used to generate it.

As described in Fig. 4, to identify the vehicle we apply the same algorithm HMAC with SHA512, but in this case, the requested necessary information can be retrieved directly from the vehicle. We decided to use the Vehicle Identification Number (VIN), a unique 17 digits code defined in its format by ISO 3779:2009 [57] and used by the automotive industry to identify unequivocally and worldwide each vehicle. Starting from 1981, VIN has a common format that explains several characteristics of the vehicle, but in our research, the main characteristics is that is unique worldwide, so we can use it as our input in HMAC, while as secret we can use a random password generated in a safe way and environment by the Trusted Platform Module (TPM) of the vehicle. Since the password can be stored directly in the vehicle and that the user does not need to know it, the TPM can generate and store a complex, random and long password, more than ePSW that it can assure the safety of the digest from brute-force attacks. For these reasons, we applied HMAC with SHA-512 with VIN as input and the TPM generated password as secret, generating a unique digest of 64 bytes.

Applying the described procedures (Fig. 4), we generate two digests of 64 bytes each: the first identifying the driver, while the second identifying the vehicle. The concatenation of the two digests generates the DSI. Thus, the E-Corridor infrastructure can identify each user with an unequivocally and deterministic code, identifying in the first 128 characters the driver and in the next 128 characters the vehicle. In the case of car sharing, the first 128 characters will change but the next 128 will be the same.

We can state that this DSI generation process is privacy-preserving, because it allows users to know clearly which personal information is used to generate the code and it protects the confidentiality of the data. Besides, with this process, we have generated a unique identifier that allows users to receive services from the E-Corridor infrastructure.

VI. Experimental Evaluation

In this section, we show the feasibility of the privacy-preserving Driver DNA analytic within the E-Corridor

infrastructure. In particular, we illustrate how drivers will be able to obtain customized services by not disclosing out sensitive data. To this purpose, we show the feasibility of the privacy-preserving Driver DNA analytic using the ORE method as privacy encryption. Then, we compare the ORE method with the HE to maintain drivers' privacy when establishing the Driver DNA. Both, HE and ORE directly work on encrypted data, so we compare the two solutions to find the most suitable for our architecture.

A. Involved Hardware

All the operations and computations were performed with a notebook with a processor Intel(R) Core (TM) i5-1035G1 CPU 1.00 GHz with turbo max frequency 3.60 GHz and 8 GB installed RAM. The considered operations and computations span from the dataset generation, to the implementation of the Driver DNA analytic using both ORE and HE.

B. Dataset Generation

To evaluate the reference architecture in Fig. 1, we simulate a road environment composed by drivers that in cars ask for a service such as the price of fuel. We select this service since refueling represents on the most frequent and common situations for a driver and it allows us to use one of the of the four DNA parameters: the engine RPM. We consider a vehicle in a simulated town environment and we retrieve Driver DNA data in real-time to create a dataset to perform analytics when required.

Compared with also other simulator, such as AirSim [58], PGDrive [59] and SUMMIT [60], for our experiments we chose CARLA *(CAR Learning to Act)*, an open-source simulator for autonomous driving research, since it is vehicle parameters centered and highly customizable for every element of the road infrastructure with different available environments and vehicles. CARLA is grounded to simulate on Unreal Engine [61], a complete suite of creation tools to simulate real-time application, and uses the OpenDRIVE standard to define roads and urban settings [62]. CARLA simulates realistic weather conditions and advanced urban scenarios using different types of vehicles, equipped with customizable sensors, like cameras, LIDAR, radar, GNSS, and inertial measurement units. Besides, it is possible to retrieve some vehicle parameters like speed, steering, throttle, and brake usage.

After the definition of the test conditions as shown in Table I, we start our simulation using the CARLA autopilot function that drives the car in the urban environment, respecting traffic rules and without a specific destination. In the autopilot mode, CARLA offers the possibility to personalize some aspects of the vehicle and the driving style. In our scenario, we do not modify any default parameter, because we are only interested in the engine RPM value to test random driving styles. The cars are driven in the town without any particular default journey or target. It is simulated only a specific time slot driving in a traffic environment with other cars, pedestrians, speed limits, traffic lights, and crossroads.

In our simulations, we focus on the engine RPM values to calculate the driver DNA to provide customized fuel price.

TABLE I
EXPERIMENT CONDITIONS

Driving simulator:	CARLA [9]
API language:	Python / C++
Driver:	CARLA Autopilot
Driving environment:	Urban
Weather conditions:	Sunny
Vehicle A:	Audi A2
Vehicle B:	Citroen C3
Vehicle C:	Mini Cooper
Vehicle D:	Nissan Micra
Vehicle F:	Seat Leon
Vehicles power supply:	fuel
Non-player characters:	Yes (vehicles and walkers)
Simulation duration:	3 slots of 20 minutes each
Retrieved value:	Engine RPM

Firstly, we define the environment in which we performed the simulation. We use the CARLA default environment *Town05*, a squared-grid town with cross junctions, a bridge, and multiple lanes per direction. We consider this town as a closed circuit, where cars moved, like in a single road segment. In this environment, we generate randomly with CARLA several vehicles and walkers to simulate a real drive in the traffic with non-player characters. As weather conditions, we choose a sunny day without any particular climate event since, in our test, we need only to retrieve the engine RPM values. Thus, we do not consider the weather a significant variable. As vehicles, we test five fuel-powered cars from near car market segments like a mini car (B) or compact car (C), chosen from the available CARLA default vehicles. For our experiments, the only constraint on the vehicle is to choose fuel-powered vehicles and not electric ones, because the engine RPM values are less significant for an Electric Vehicle (EV), since EV can generate about the same amount of torque at the minimum engine RPM value as at the maximum engine RPM value.

To avoid any bias deriving from autopilot, for every vehicle, we simulate three time slots of twenty minutes driving each. This choice allows us to retrieve on average 1215 values/hour with an engine RPM value every 2.9 seconds. With CARLA, we create three different datasets for each of the five cars, containing the engine RPM values retrieved in the twenty minutes of driving. The next goal is to calculate Driver DNA, by means of the average operation to identify to which quantile, with respect to Service Provider database, the value belongs. As stated before, the different simulations are performed to identify possible biases, caused by autopilot, but we do not report any particular deviations.

C. Driver DNA Using ORE

The mean of the engine RPM values has to be sanitized before sending to the edge node. We use the hash function SHA-512 [54] that generates a 64-bites digest. If the original plaintext is large, it is almost unfeasible to recover the original

TABLE II
RPM MEAN AND RANK FOR EACH DRIVING SLOT

	Slot 1		Slot 2		Slot 3	
	Mean	*Rank*	*Mean*	*Rank*	*Mean*	*Rank*
Vehicle A:	1558	3	1588	3	1570	3
Vehicle B:	1448	2	1465	2	1486	2
Vehicle C:	1587	3	1621	3	1571	3
Vehicle D:	3084	5	2981	5	2945	5
Vehicle F:	1347	3	1388	3	1440	3

message from the digest and it provides strong resistance to collision and preimage attacks. In our example, to avoid any possible malicious attack from a rainbow table, which is a precomputed table containing all possible outputs of cryptographic hash functions, we decide to add a secret random large number to engine RPM mean and to quantile limits. This secret number has to be defined and shared before in the E-Corridor infrastructure among the vehicle and the SP.

Before calculating the Driver DNA, we have to define the quantile limits. As stated in Section V, the quantile limits should be defined by the SP according to its proprietary policies. A company can establish and modify the limits following, for example, the evolution of its database. For instance, if most parts of drivers are in a specific interval, the service provider can decide to split this interval into more parts or vice versa in the opposite case. In our research, the quantile limits are not significant and they have to be established only to be a base for our tests. For this reason, we decide to have 5 generic and conceivable intervals with the following ranges: [0-1000][1001-1500][1501-2000][2001-2500][2501-3000].

Then, we create a specific program in Python language [63] to sanitize with SHA-512 method the retrieved mean and each quantile limit. After these operations, the edge node receives the sanitized data. To perform the comparison between the mean and the quantile limits, we implement in Python the ORE method, following the Chenette schema [44], and we identify for each driving slot to which quantile the mean belongs.

In Table II, we report the retrieved results. For each vehicle and each driving slot, the value that defines the fuel price is reported, i.e., rank value. In our scenario, the higher is the rank value the higher is higher the price. On the opposite situation, the lower rank value indicates a lower price. The SP will receive this rank value from the edge node without knowing anything in addition to the driver's data. Finally, the computation time of ORE comparisons to identify the right quantile has been quantified. We tested ten times every comparison with the ORE method and the average time was about 1.5 milliseconds to identify to which quantile the mean belonged.

D. Driver DNA Using Homomorphic Encryption

Here, we adopt the Homomorphic Encryption (HE) method to preserve the data privacy again when running the Driver DNA analytic. HE is a cryptosystem that supports computation on encrypted data and enables outsourcing of data storage and analysis [64]. Moreover, nowadays it is possible to implement

TABLE III
FIRST TEST SET VALUES TO IDENTIFY MAXIMUM NUMBER OF VALUES THAT CAN BE PROCESSED TOGETHER

Number values	**Time in s**	**Sum correct**	**Average correct**
2	> 0.00	Yes	Yes
10	0.01	Yes	Yes
100	0.10	Yes	Yes
200	0.20	Yes	Yes
250	0.23	Yes	No
300	0.28	Yes	No
350	0.32	Yes	No
400	0.35	Yes	No
450	0.43	Yes	No
500	0.46	Yes	Yes
550	0.54	Yes	No
600	0.57	Yes	No

Fully Homomorphic Encryption (FHE) that allows evaluation of arbitrary functions on encrypted data like addition or multiplication. HE allows us to share data in an untrusted environment like our out-of-vehicle. We use one of the most mature and well-defined HE schemes like CKKS, which supports approximate computations of real numbers [31].

For this evaluation, we adopted the Python library *Pyfhel* (Python For Homomorphic Encryption Libraries) that implements functionalities of multiple HE libraries such as addition, multiplication, exponentiation, or scalar product in Python [65]. Moreover, Pyfhel implements the backend of the SEAL library, the most complete HE C++ library [66], [67], that supports different calculation schema with varied features. To operate both on integer and float numbers, we use the CKKS scheme. However, we have to point out that as reported in [68], the CKKS schema can yield some approximate results especially with a large number of performed operations. To deal with this problem, we performed several tests to choose the correct setting. This allows us to maintain an acceptable approximation to verify the scalability of the HE method, i.e., understanding the limit related to the number of values that can be processed by Pyfhel.

To verify the scalability of the HE method, we evaluated its limits by computing the average of a generic set $A = a_1,...,a_i$ of n elements, since HE supports only addition and multiplication, we multiply the sum for $1/n$, instead of performing division.

We tested the parameters in Table III that shows in the first column the number of values used to compute the average. The second column shows the time needed to encrypt the values, compute the average and decrypt the results with different input dataset sizes. It should be noted that the computational time grows constantly at the increase of the size of the dataset. The third column reports if the sum of all elements was properly performed on the encrypted data. In the fourth column, we report if the multiplication, and so the average, is rightly calculated.

To summarize Table III, we observe that Pyfhel can properly work using up to 200 elements as input when establishing the Driver DNA.

VII. Privacy Metrics

In the following two Sections, VII-A and VII-B, we compute the *privacy gain* and the *data loss utility* to evaluate privacy-preserving properties using the data generated by the ORE and HE process.

A. Privacy Gain

To verify if ORE and HE methods are privacy-preserving, we define a metric to measure the privacy gain on sanitized data. We decide to use Shannon Entropy (SE) as defined in [69] and applied in [47] and [70]. SE quantifies the amount of information in a variable. If the information contained in a message is surprising, we obtain a higher value of entropy. Instead, if the message is not surprising, but its composition is expected, we obtain a lower value of entropy. The mathematical definition for a random variable X is Equation 3, where $p(x)$ is the probability mass function of X and Σ denotes the sum over the variable's possible values. The choice of the base of the logarithm can vary according to different applications and the minus before the sum is necessary to have a meaningful and non-negative entropy. In fact, probability density value $p(x)$ lays in the range from 0 to 1, which means that logarithm functions will take on a negative value.

$$Entropy\ H(X) = -\sum_{i=1}^{n} p(x_i) \log p(x_i) \tag{3}$$

After the computation of the entropy of the two values before and after the sanitization, we defined privacy-preserving value (pp) in Equation 4. If pp value is less than 1, the entropy after the application of privacy mechanism is decreased, so we have a loss of privacy. If pp value is equal to 1 means that the privacy gain is *0*. If pp is more than 1, it means that we have a privacy gain caused by the increase of the entropy between the two datasets before and after the application of privacy mechanisms.

$$Privacy\_Preserving\ (pp) = \frac{H(after)}{H(before)} \tag{4}$$

To obtain a privacy-preserving value we need to compute the entropy on the value before and after the encryption. The raw value of the mean is retrieved from the dataset that we created in §VI with CARLA simulator, while the sanitized values of ORE and HE have been respectively generated applying SHA-512 method, before ORE comparison and applying directly the HE method with the Pyfhel library.

With these data, we can quantify the gain of privacy as defined in Equation 4. We apply the function on each retrieved RPM mean for each vehicle. Moreover, we calculate the privacy gain for the quantile limits. This information is sent from the SP to the EN to retrieve in which quantile the RPM average belongs, so they should be protected. The privacy gain results are reported in Table IV.

From Table IV, we can observe a significant privacy gain after the application of ORE or HE. All the values, both for RPM and each limit, are larger than 1. This means an increase in privacy after the protection process. Hence, data that are sent from the vehicle to the edge node have an increase in privacy, confirming our privacy-preserving process.

TABLE IV
Privacy Gain After Application of Privacy Mechanism

***Privacy gain*: < 1 loss; = 1 no loss or gain; > 1 gain**						
QL = Quantile Limit						
Vehicle	**RPM value**	**Entropy before**	**Entropy after ORE**	**Entropy after HE**	**Privacy Gain ORE**	**Privacy Gain HE**
A	1558	1.04	2.29	3.57	2.20	3.44
	1588	1.04	2.26	3.57	2.18	3.44
	1570	1.38	2.27	3.56	1.64	2.56
B	1448	1.04	2.25	3.56	2.16	3.42
	1465	1.39	2.26	3.50	1.63	2.52
	1486	1.38	2.25	3.54	1.62	2.55
C	1587	1.39	2.28	3.55	1.65	2.56
	1621	1.04	2.27	3.55	2.19	3.42
	1571	1.04	2.26	3.58	2.18	3.44
D	3084	1.39	2.27	3.53	1.64	2.55
	2981	1.39	2.28	3.54	1.65	2.55
	2945	1.39	2.26	3.53	1.63	2.55
E	1347	1.39	2.28	3.56	1.64	2.57
	1388	1.04	2.29	3.55	2.20	3.42
	1440	1.04	2.26	3.55	2.18	3.41
QL	0	0.00	2.25	3.58	inf	inf
QL	1000	0.56	2.27	3.60	4.04	6.30
QL	1500	1.03	2.28	3.50	2.19	3.36
QL	2000	0.56	2.25	3.54	4.01	6.29
QL	2500	1.03	2.26	3.51	2.18	3.38
QL	3000	0.56	2.25	3.55	4.01	6.32

B. Data Loss Utility

After defining the privacy gain, we quantify the data loss utility, defined as *the amount of information that we lost after the application of privacy methods* [47]. In particular, there is always a trade-off between the risk of disclosure and data utility. To preserve privacy, if we modify too much the original raw dataset, it may become impossible to perform correct analytics on sanitized data. On the other side, if the sanitization of original raw data is low, there could be a disclosure of sensitive information. For this reason, we define which analytics can be still performed correctly on our sanitized dataset after the application of ORE or HE. As suggested by [47], to measure the remaining data utility after encryption, we could apply a non-symmetric measure between two distributions called Kullback-Leibler divergence (KL divergence) [71]. This measure studies two different distributions using its easy optimization and its maximum likelihood estimation, even if it does not take into account how close two outcomes might be, but only their relative probability [72]. This last limit does not affect our study since we only need to find the relative probability that an attacker may retrieve the raw data from encrypted data, so we do not need to know the general distance between the two distributions. Following, we compute KL divergence on our dataset before and after encryption with ORE and HE. In our study, during the KL divergence computation, we notice that

TABLE V
DATA LOSS UTILITY - POSSIBLE OPERATIONS

Operation	ORE	HE Pyfhel
Ciphertexts Comparison	Yes	No
Addition ciphertexts	No	Yes
Multiplication ciphertext per int/float	No	Yes
Multiplication between ciphertexts	No	Yes

it is not significant to compare the distribution of raw data with distribution of sanitized data, which does not follow a distribution derived from raw data like it may be in the differential privacy. Hence, we search for another solution to compute data loss utility.

Several options, e.g., [73], [74], were proposed to compute information loss after anonymization. The proposed solutions use comparisons between the raw data and the encrypted data using their distributions or correlation matrix. With the proposed approaches, we may not retrieve any significant value because a direct comparison between raw and sanitized data could be again meaningless. However, in our architecture, we state that we do not have data loss utility if we are able to perform all the necessary operations to enable users to receive customized services. For this purpose, we need that the encryption method is able to correctly perform comparisons between ciphertext. As shown in Table V, ORE method is able to perform these operations, while HE may perform other operations, but not the comparison between ciphertexts without interaction with the subject to share a secret key.

To conclude, the information loss on sanitized data with ORE process can be determined from the right result of the comparisons.

VIII. DISCUSSION ON ORE - HE METHODS FOR PRIVATE DRIVER DNA

We apply both ORE and HE methods on the same dataset and process to compute the privacy gain as shown in Table IV. In particular, we can notice that with HE we can obtain a higher privacy gain than with ORE. This is a positive feature of the HE method, but the privacy gain generated with ORE, as shown before, does not allow an attacker to steal personal information from data. So, we consider this finding enough to keep drivers' data private when running the Driver DNA analytic.

The above comparison between ORE ad HE is the only positive finding that we can link to HE. In the following, we list other findings of this comparison in which HE does not overtake the ORE method on when running the analytic.

- On an input of maximum 200 values, the computation time of HE is much larger than the computation time of ORE. Besides, at the increase of the input size, also over 200 inputs, the difference continues to grow as shown in Fig. 5. In a time-sensitive infrastructure like VANET, timing is an important element and HE is too slow.
- As shown before, in our architecture, HE shows the limit of 200 inputs, while ORE does not have these limitations. Considering that a vehicle can generate more than one input per second for each of the four parameters of driver DNA, HE may need to compute the metric every minute with a higher computational cost, while ORE can be applied on a larger dataset.
- HE does not support comparison between ciphertexts with Pyfhel, so it is necessary to decrypt data on the Edge Node of E-Corridor to make comparisons. In this way, data are clear on the Edge Node, which can be considered trusted, but with ORE this decryption operation is not necessary. ORE can compare encrypted data, so it may be considered more secure for our infrastructure.
- HE requires the usage of secret keys in its protocol, while ORE can work without shared secret keys.

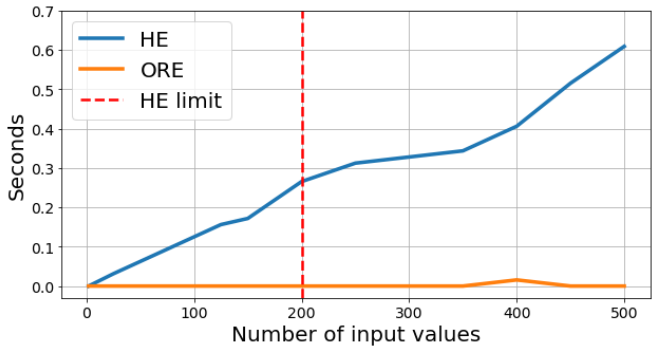


Fig. 5. Elapsed time of HE and ORE to compute the quantile of the user's driver DNA as the number of input values increases.

To conclude, considering the above motivations, we confirm that the ORE method is more suitable for our infrastructure than HE since it is faster, it works with a large amount of data and it is more efficient in comparison operations than HE. On the other side, HE can be considered a good candidate for other infrastructures, where more operations like additions or multiplications on encrypted data are needed, but only if it is possible to decrease the computation time.

IX. PRIVACY-PRESERVING SCHEMA COMPARISON

After the definition of ORE as the best method for our infrastructure, we decide to compare our schema, which we call ORE, with other existing privacy-preserving schemas. As described in §II, our work can be classified in the Identity Based Authentication (IBA) category, where the identifier is derived from driver and vehicle information. For this reason, we compare our schema with the schemas of the same category as described in [19] and [21]. Our schema uses the ORE method to provide the service and it does not need any other operation, except for the message exchange. For this reason, even if our schema can be classified in the IBA category, we can not compare it with the other bilinear pairing schemas or use performance metrics because the only operations that ORE implements are the numeric comparison. The only possible comparison with other schemas is the feature comparison as reported in the survey [21], published in 2021, of Mundhe et al. which reports several IBA schemas and lists their advantages and limitations. Table VI, derived from Table V of Mundhe's survey [21], reports if our schema keeps the advantages, but also it overcomes the limitations of some of the current IBA schemas.

TABLE VI
SCHEMA COMPARISON BASED ON TABLE V OF MUNDHE'S SURVEY [21]

Features	Schemas												
	[77]	[78]	[79]	[80]	[75]	[81]	[82]	[83]	[84]	[76]	[85]	[86]	ORE
Properties and advantages													
Privacy-preserving authentication	✓	✓	✓	×	✓	×	×	✓	×	✓	×	×	✓ [DSI identifier, ORE schema, encrypted messages]
It entirely does not depend upon trusted dealers and operates with efficient key updation	×	×	×	✓	×	×	×	×	×	×	×	×	✓ [We do not have a TA]
It provides message authentication and unforgeability	×	×	×	×	✓	×	×	×	×	×	×	×	×
An RSU can verify the group of signatures received from many vehicles at the same time	×	×	×	×	×	✓	×	×	×	×	×	×	✓ [RSU can verify multiple vehicle at the same time]
The TA pre-loads the real identity, password, and private keys into each vehicle	×	×	×	×	×	×	✓	×	×	×	×	×	✓ [The DSI is calculated and stored in the vehicle]
This scheme performs an efficient revocation of vehicles using the identity-based privacy-preserving authentication	×	×	×	×	×	×	×	×	✓	×	×	×	✓ [The E-Corridor infrastructure can revoke a DSI]
It provides message traceability, unlinkability, and non-repudiation	×	×	×	×	×	×	×	×	×	✓	×	×	×
This scheme uses identity and HMAC-based hybrid cryptography to secure vehicular communications	×	×	×	×	×	×	×	×	×	×	✓	×	✓ [HMAC for the identifier and encryption methods for message exchange]
This scheme allows a cloud user to restrict where the data can be processed using a new location based encryption scheme	×	×	×	×	×	×	×	×	×	×	×	✓	✓ [Data are processed in the vehicle and sent only in encrypted form (ORE method)]
Limitations													
No mathematical proof or graph to demonstrate the performance	✓	×	✓	×	✓	×	✓	×	✓	✓	×	×	× [We compare ORE and HE to show the performance of our methods (§VIII and we evaluate the maximum quantity of data that can be processed at the same time (Table III)]
The complexity is high because of separate mutual authentication mechanism for vehicles and RSUs	×	✓	×	×	×	×	×	×	×	×	×	×	× [Authentication mechanism are not separate and the complexity of ORE schema is low without key exchanges or TA]
Vehicles need to use the same group private key to encrypt and decrypt the messages	×	×	×	✓	×	×	×	×	×	×	×	×	× [Each vehicle and driver has its own keys and identifier]
The key escrow problem can dismantle the entire network as the TA can compute all the vehicle's private keys	×	×	×	×	×	✓	×	×	×	×	×	×	× [E-Corridor infrastructure does not know the values of the DSI components, but only the DSI value]
It cannot provide conditional privacy and batch message verification	×	×	×	×	×	×	×	✓	×	×	×	×	✓ [ORE does not provide message verification]
It does not consider the batch verification of messages and signatures	×	×	×	×	×	×	×	×	×	×	✓	×	✓ [ORE does not provide message verification]
It is difficult to broadcast messages if vehicles are sparsely located	×	×	×	×	×	×	×	×	×	×	×	✓	× [ORE does not suffer this issue because the service is a single vehicle-provider so it does not need any other vehicle or group signatures]

We can notice that our schema, identified as ORE in Table VI, keeps most of the advantages of the other schemas. Although, ORE does not provide unforgeability, traceability, and non-repudiation as the schemas [75] and [76], it contains other advantages, such as: i) a privacy-preserving authentication schema, ii) the overcome of the issues derived from

an external TA, and iii) the process of data in clear only inside the vehicle. In the second part of Table VI we show the limitations of the compared schema, and we can notice that ORE has only two limitations, both related to the lack of message verification. The other limitations are defeated by ORE as described in Table VI.

To conclude, the ORE schema has most of the advantages of the current IBA schemas and it can provide a significant and efficient improvement. However, ORE is designed in a defined infrastructure and context. The application of ORE in different scenarios may request to modify some components and add some solutions, for example, for message authentication.

X. Conclusion and Future Work

Our research defines a privacy-preserving architecture where a driver can receive personalized services, based on driving style. We describe our solution, starting from the European project E-Corridor, we identify the driving style mechanism, Driver DNA, based on four different vehicles parameters, and we assure the privacy using ORE mechanism. To show the advantages of ORE we compare our method with HE, and, then, we compare our schema with some current schemas for VANETs.

The results of our research show the possibility to receive personalized services from ITS, but, at the same time, preserving driver's privacy. Our work can be a baseline for further studies on this topic and it could be a great chance to incentivize drivers to have a specific driving attitude to improve safety on the roads and to save the environment by reducing vehicle fuel consumption.

In the future several possible implementations can be achieved. For example, the use of HE to compare ciphertexts can increase the possibilities of our infrastructure, which is actually limited to comparison operations with the ORE method. Another possible encryption method that can be applied is secure Multi-Party Computation (MPC), which allows peers to share data and preserve privacy. The infrastructure can be also tested using data retrieved from real vehicles, while in this research we simulated data with CARLA software. Besides, driver DNA, which is actually composed of four parameters, can be extended using more suitable data for electric vehicles or it can be applied on autonomous vehicles to define a specific algorithm for driving style. Finally, our infrastructure can be studied with the data-sharing agreements (DSA) among the driver and the different road actors like E-Corridor infrastructure or the service providers.

References

[1] E. A. Taub. (Mar. 2021). *Carmakers Strive to Stay Ahead of Hackers*. New York. [Online]. Available: https://www.nytimes.com/2021/03/18/business/hacking-cars-cybersecurity.html

[2] J. Wang, Y. Liu, S. Niu, and H. Song, "Reinforcement learning optimized throughput for 5G enhanced swarm UAS networking," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Montreal, QC, Canada, Jun. 2021, pp. 1–6, doi: 10.1109/ICC42927.2021.9500733.

[3] *The European Parliament and the Council*. Directive 2010/40/eu. Accessed Mar. 2022. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX%3A32010L0040

[4] Consiglio Nazionale delle Ricerche. *E-Corridor*. Accessed: Oct. 28, 2022. [Online]. Available: https://e-corridor.eu/

[5] Y. Lei, K. Liu, Y. Fu, X. Li, Z. Liu, and S. Sun, "Research on driving style recognition method based on drivers dynamic demand," *Adv. Mech. Eng.*, vol. 8, no. 9, Sep. 2016, Art. no. 1687814016670577.

[6] G. Costantino, F. Martinelli, I. Matteucci, and P. Santi, "A privacy-preserving infrastructure for driver's reputation aware automotive services," in *Proc. Int. Workshop Socio-Tech. Aspects Secur. Trust*, in Lecture Notes in Computer Science, vol. 11739. Luxembourg: Springer, Sep. 2019, pp. 159–174.

[7] U.S. National Institute of Standards and Technologies (NIST). *Glossary*. Accessed: Oct. 28, 2022. [Online]. Available: https://csrc.nist.gov/glossary/term/authenticity

[8] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. 1st Annu. Conf. Robot Learn.*, 2017, pp. 1–16.

[9] *Connected Cars*. Accessed: Mar. 2022. [Online]. Available: https://connectedcars.io/

[10] *Drive Quant*. Accessed: Mar. 2022. [Online]. Available: https://www.drivequant.com/

[11] *Zenroad*. Accessed: Mar. 2022. [Online]. Available: https://www.damoov.com/telematics-app/

[12] *Mobile Apps and Stealing a Connected Car*. Accessed: Mar. 10, 2022. [Online]. Available: https://securelist.com/mobile-apps-and-stealing-a-connected-car/77576/

[13] A. Dorri, M. Steger, S. S. Kanhere, and R. Jurdak, "BlockChain: A distributed solution to automotive security and privacy," *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 119–125, Dec. 2017, doi: 10.1109/MCOM.2017.1700879.

[14] V. H. Le, J. den Hartog, and N. Zannone, "Security and privacy for innovative automotive applications: A survey," *Comput. Commun.*, vol. 132, pp. 17–41, Nov. 2018, doi: 10.1016/j.comcom.2018.09.010.

[15] L. Zhou, S. Du, H. Zhu, C. Chen, K. Ota, and M. Dong, "Location privacy in usage-based automotive insurance: Attacks and countermeasures," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 1, pp. 196–211, Jan. 2019, doi: 10.1109/TIFS.2018.2848227.

[16] B. Nelson and T. Olovsson, "Introducing differential privacy to the automotive domain: Opportunities and challenges," in *Proc. IEEE 86th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2017, pp. 1–7, doi: 10.1109/VTCFall.2017.8288389.

[17] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, Jul./Oct. 1948.

[18] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951, doi: 10.1214/aoms/1177729694.

[19] S. A. Jan, N. U. Amin, M. Othman, M. Ali, A. I. Umar, and A. Basir, "A survey on privacy-preserving authentication schemes in VANETs: Attacks, challenges and open issues," *IEEE Access*, vol. 9, pp. 153701–153726, 2021.

[20] A. K. Malhi, S. Batra, and H. S. Pannu, "Security of vehicular Ad-hoc networks: A comprehensive survey," *Comput. Secur.*, vol. 89, Feb. 2020, Art. no. 101664. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404818312872

[21] P. Mundhe, S. Verma, and S. Venkatesan, "A comprehensive survey on authentication and privacy-preserving schemes in VANETs," *Comput. Sci. Rev.*, vol. 41, Aug. 2021, Art. no. 100411, doi: 10.1016/j.cosrev.2021.100411.

[22] X. Li et al., "PAPU: Pseudonym swap with provable unlinkability based on differential privacy in VANETs," *IEEE Internet Things J.*, vol. 7, no. 12, pp. 11789–11802, Dec. 2020.

[23] J. Wang, Y. Liu, S. Niu, and H. Song, "Lightweight blockchain assisted secure routing of swarm UAS networking," *Comput. Commun.*, vol. 165, pp. 131–140, Jan. 2021, doi: 10.1016/j.comcom.2020.11.008.

[24] Y. Liu et al., "Blockchain enabled secure authentication for unmanned aircraft systems," 2021, *arXiv:2110.08883*.

[25] X. Feng, Q. Shi, Q. Xie, and L. Liu, "An efficient privacy-preserving authentication model based on blockchain for VANETs," *J. Syst. Archit.*, vol. 117, Aug. 2021, Art. no. 102158, doi: 10.1016/j.sysarc.2021.102158.

[26] Q. Feng, D. He, S. Zeadally, and K. Liang, "BPAS: Blockchain-assisted privacy-preserving authentication system for vehicular ad hoc networks," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4146–4155, Jun. 2020.

[27] Y. Ren, W. Liu, A. Liu, T. Wang, and A. Li, "A privacy-protected intelligent crowdsourcing application of IoT based on the reinforcement learning," *Future Gener. Comput. Syst.*, vol. 127, pp. 56–69, Feb. 2022, doi: 10.1016/j.future.2021.09.003.

[28] L. Wang, G. Liu, and L. Sun, "A secure and privacy-preserving navigation scheme using spatial crowdsourcing in fog-based VANETs," *Sensors*, vol. 17, no. 4, p. 668, Mar. 2017. [Online]. Available: https://www.mdpi.com/1424-8220/17/4/668

[29] Q. Kong, R. Lu, M. Ma, and H. Bao, "A privacy-preserving and verifiable querying scheme in vehicular fog data dissemination," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1877–1887, Feb. 2019.

[30] W. Qinglong, T. Zhiqiang, F. Na, and D. Zongtao, "A privacy preserving data collecting scheme in vanet," in *Proc. Green, Pervas., Cloud Comput. GPC Workshops*, J. Wang, L. Chen, L. Tang, and Y. Liang, Eds. Singapore: Springer, 2020, pp. 45–58.

[31] J. H. Cheon, A. Kim, M. Kim, and Y. S. Song, "Homomorphic encryption for arithmetic of approximate numbers," in *Proc. Adv. Cryptol. 23rd Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, in Lecture Notes in Computer Science, vol. 10624, T. Takagi and T. Peyrin, Eds. Hong Kong: Springer, Dec. 2017, pp. 409–437, doi: 10.1007/978-3-319-70694-8_15.

[32] J. Zhang, H. Zhong, J. Cui, M. Tian, Y. Xu, and L. Liu, "Edge computing-based privacy-preserving authentication framework and protocol for 5G-enabled vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7940–7954, Jul. 2020, doi: 10.1109/TVT.2020.2994144.

[33] Z. Wei, J. Li, X. Wang, and C.-Z. Gao, "A lightweight privacy-preserving protocol for vanets based on secure outsourcing computing," *IEEE Access*, vol. 7, pp. 62785–62793, 2019.

[34] R. S. Jurecki and T. L. Stanczyk, "A methodology for evaluating driving styles in various road conditions," *Energies*, vol. 14, no. 12, p. 3570, Jun. 2021.

[35] S. Lefevre, A. Carvalho, Y. Gao, E. Tseng, and F. Borrelli, "Driver models for personalized driving assistance," *Vehicle Syst. Dyn.*, vol. 53, pp. 1705–1720, Jul. 2015.

[36] J. Carmona, F. García, D. Martín, A. de la Escalera, and J. M. Armingol, "Data fusion for driver behaviour analysis," *Sensors*, vol. 15, no. 10, pp. 25968–25991, Oct. 2015, doi: 10.3390/s151025968.

[37] M. Enev, A. Takakuwa, K. Koscher, and T. Kohno, "Automobile driver fingerprinting," *Privacy Enhancing Technol.*, vol. 2016, no. 1, pp. 34–50, 2016, doi: 10.1515/popets-2015-0029.

[38] M. Van Ly, S. Martin, and M. M. Trivedi, "Driver classification and driving style recognition using inertial sensors," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2013, pp. 1040–1045.

[39] S. Zeng and Y. Chen, "Concurrently deniable group key agreement and its application to privacy-preserving VANETs," *Wirel. Commun. Mob. Comput.*, vol. 2018, Apr. 2018, Art. no. 6870742, doi: 10.1155/2018/6870742.

[40] S. S. Moni and D. Manivannan, "Privacy-preserving and Authentication in VANETs," *Int. J. Next Gener. Comput.*, vol. 11, no. 2, pp. 98–124, 2020. [Online]. Available: http://perpetualinnovation.net/ojs/index.php/ijngc/article/view/531

[41] National Human Genome Research Institute (NHGRI). *Glossary*. Accessed: Oct. 28, 2022. [Online]. Available: https://www.genome.gov/genetics-glossary/Deoxyribonucleic-Acid

[42] E. Massaro et al., "The car as an ambient sensing platform," *Proc. IEEE*, vol. 105, no. 1, pp. 3–7, 2017, doi: 10.1109/JPROC.2016.2634938.

[43] U. Fugiglando, P. Santi, S. Milardo, K. Abida, and C. Ratti, "Characterizing the 'driver DNA' through can bus data analysis," in *Proc. 2nd ACM Int. Workshop Smart, Auto., Connected Veh. Syst. Services*. New York, NY, USA: Association for Computing Machinery, 2017, pp. 37–41, doi: 10.1145/3131944.3133939.

[44] N. Chenette, K. Lewi, S. A. Weis, and D. J. Wu, "Practical order-revealing encryption with limited leakage," in *Fast Software Encryption* (Lecture Notes in Computer Science), vol. 9783, T. Peyrin, Ed. Bochum, Germany: Springer, Mar. 2016, pp. 474–493, doi: 10.1007/978-3-662-52993-5_24.

[45] A. J. Paverd and A. C. Martin, "Modelling and automatically analysing privacy properties for honest-but-curious adversaries," Univ. Oxford, Oxford, U.K., 2014.

[46] E-Corridor. (2021). *Resources*. [Online]. Available: https://e-corridor.eu/resources/

[47] F. Martinelli, A. Saracino, and M. Sheikhalishahi, "Modeling privacy aware information sharing systems: A formal and general approach," in *Proc. IEEE Trustcom/BigDataSE/ISPA*, Tianjin, China, Aug. 2016, pp. 767–774, doi: 10.1109/TrustCom.2016.0137.

[48] A. Ometov, S. Bezzateev, N. Mäkitalo, S. Andreev, T. Mikkonen, and Y. Koucheryavy, "Multi-factor authentication: A survey," *Cryptography*, vol. 2, no. 1, p. 1, Jan. 2018, doi: 10.3390/cryptography2010001.

[49] T-Group. (2021). *Digital Driver's License—Your id in Your Smartphone*. [Online]. Available: https://www.thalesgroup.com/en/markets/digital-identity-and-security/government/driving-licence/digital-driver-license

[50] *Personal Identification—Iso-Compliant Driving Licence—Part 5: Mobile Driving Licence (MDL) Application*, International Organization for Standardization, Geneva, CH, USA, Standard ISO/IEC FDIS 18013-5, 2021. [Online]. Available: https://www.iso.org/standard/69084.html

[51] M. Yildirim and I. Mackie, "Encouraging users to improve password security and memorability," *Int. J. Inf. Secur.*, vol. 18, no. 6, pp. 741–759, Dec. 2019, doi: 10.1007/s10207-019-00429-y.

[52] H. Krawczyk, M. Bellare, and R. Canetti, *HMAC: Keyed-Hashing for Message Authentication*, document RFC 2104, 1997, pp. 1–11, doi: 10.17487/RFC2104.

[53] McAfee. (2021). *U.S. Driver's License Numbers*. [Online]. Available: https://success.myshn.net/Policy/Data_Identifiers/U.S._Driver's_License_NumbersGeneral_Driver's_License_Keywords

[54] T. Hansen and DEE-3rd, *U.S. Secure Hash Algorithms (SHA and HMAC-SHA)*, document RFC 4634, Aug. 2006. [Online]. Available: https://rfc-editor.org/rfc/rfc4634.txt

[55] *U.S. Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)*, document RFC 6234, May 2011. [Online]. Available: https://rfc-editor.org/rfc/rfc6234.txt

[56] H. Cheng, D. Dinu, and J. Großschädl, "Efficient implementation of the SHA-512 hash function for 8-bit AVR microcontrollers," in *Innovative Security Solutions for Information Technology and Communications*, J.-L. Lanet and C. Toma, Eds. Cham, Switzerland: Springer, 2019, pp. 273–287.

[57] *Road Vehicles—Vehicle Identification Number (VIN)—Content and Structure*, International Organization for Standardization, Geneva, CH, USA, Standard ISO 3779:2009, 2009. [Online]. Available: https://www.iso.org/standard/52200.html

[58] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "AirSim: High-fidelity visual and physical simulation for autonomous vehicles," 2017, *arXiv:1705.05065*.

[59] Q. Li, Z. Peng, Q. Zhang, C. Liu, and B. Zhou, "Improving the generalization of end-to-end driving through procedural generation," 2020, *arXiv:2012.13681*.

[60] P. Cai, Y. Lee, Y. Luo, and D. Hsu, "SUMMIT: A simulator for urban driving in massive mixed traffic," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 4023–4029.

[61] Epic Games. *Unreal Engine*. Accessed: Oct. 28, 2022. [Online]. Available: https://www.unrealengine.com

[62] Association for Standardization of Automation and Measuring Systems. (2021). *Asam Opendrive*. [Online]. Available: https://www.asam.net/standards/detail/opendrive/

[63] C. D. Costa. (2020). *Top Programming Languages for Data Science in 2020*. [Online]. Available: https://towardsdatascience.com/top-programming-languages-for-data-science-in-2020-3425d756e2a7

[64] H. Chen, W. Dai, M. Kim, and Y. Song, "Efficient multi-key homomorphic encryption with packed ciphertexts with application to oblivious neural network inference," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.* New York, NY, USA: Association for Computing Machinery, Nov. 2019, pp. 395–412, doi: 10.1145/3319535.3363207.

[65] A. Ibarrondo and G. M. Laurent (SAP) Onen (EURECOM). (2021). PYthon for Homomorphic Encryption Libraries. SAP and EURECOM. [v2.3.1]. [Online]. Available: https://pyfhel.readthedocs.io/en/latest/index.html

[66] A. Viand, P. Jattke, and A. Hithnawi, "SoK: Fully homomorphic encryption compilers," in *Proc. 42nd IEEE Symp. Secur. Privacy (SP)*. San Francisco, CA, USA: IEEE, May 2021, pp. 1092–1108. Accessed: Oct. 28, 2022, doi: 10.1109/SP40001.2021.00068.

[67] S. S. Sathya, P. Vepakomma, R. Raskar, R. Ramachandra, and S. Bhattacharya, "A review of homomorphic encryption libraries for secure computation," *CoRR*, vol. abs/1812.02428, 2018. Accessed: Oct. 28, 2022. [Online]. Available: http://arxiv.org/abs/1812.02428

[68] (Nov. 2020). *Microsoft SEAL (Release 3.6)*. Microsoft Research, Redmond, WA, USA. [Online]. Available: https://github.com/Microsoft/SEAL

[69] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. Hoboken, NJ, USA: Wiley, 2006. [Online]. Available: http://www.elementsofinformationtheory.com/

[70] G. Costantino, R. Maiti, F. Martinelli, and P. Santi, "Private mobility-cast for opportunistic networks," *Comput. Netw.*, vol. 120, pp. 28–42, Jun. 2017, doi: 10.1016/j.comnet.2017.04.010.

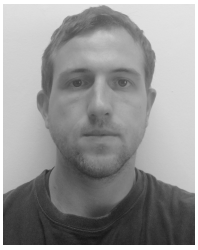
[71] J. Soch and C. Allefeld, "Kullback–Leibler divergence for the normal-gamma distribution," 2016, *arXiv:1611.01437*. [Online]. Available: https://arxiv.org/abs/1611.01437

[72] M. G. Bellemare et al., "The Cramer distance as a solution to biased Wasserstein gradients," 2017, *arXiv:1705.10743*.

[73] J. Domingo-Ferrer, J. Mateo-Sanz, and V. Torra, "Comparing SDC methods for microdata on the basis of information loss and disclosure," in *Proc. ETK-NTTS*, Jan. 2001, pp. 6–9.

[74] J. Domingo-Ferrer and V. Torra, "Disclosure control methods and information loss for microdata," in *Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies*. Amsterdam, The Netherlands: Elsevier, Jan. 2001, pp. 91–110.
[75] A. Karati, S. H. Islam, G. Biswas, M. Z. A. Bhuiyan, P. Vijayakumar, and M. Karuppiah, "Provably secure identity-based signcryption scheme for crowdsourced industrial Internet of Things environments," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2904–2914, Aug. 2018.
[76] I. Ali, T. Lawrence, and F. Li, "An efficient identity-based signature scheme without bilinear pairing for vehicle-to-vehicle communication in VANETs," *J. Syst. Archit.*, vol. 103, Feb. 2020, Art. no. 101692, doi: 10.1016/j.sysarc.2019.101692.
[77] J. Sun, C. Zhang, Y. Zhang, and Y. Fang, "An identity-based security system for user privacy in vehicular ad hoc networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 9, pp. 1227–1239, Sep. 2010, doi: 10.1109/TPDS.2010.14.
[78] H. Lu, J. Li, and M. Guizani, "A novel ID-based authentication framework with adaptive privacy preservation for VANETs," in *Proc. Comput., Commun. Appl. Conf.*, 2012, pp. 345–350.
[79] N.-W. Lo and J.-L. Tsai, "An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks without pairings," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 5, pp. 1319–1328, May 2016, doi: 10.1109/TITS.2015.2502322.
[80] L. Zhang, "OTIBAAGKA: A new security tool for cryptographic mix-zone establishment in vehicular Ad hoc networks," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 12, pp. 2998–3010, Dec. 2017, doi: 10.1109/TIFS.2017.2730479.
[81] J. Cui, J. Zhang, H. Zhong, and Y. Xu, "SPACF: A secure privacy-preserving authentication scheme for VANET with cuckoo filter," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10283–10295, 2017, doi: 10.1109/TVT.2017.2718101.
[82] S.-F. Tzeng, S.-J. Horng, T. Li, X. Wang, P.-H. Huang, and M. K. Khan, "Enhancing security and privacy for identity-based batch verification scheme in VANETs," *IEEE Trans. Veh. Technol.*, vol. 66, no. 4, pp. 3235–3248, Apr. 2017, doi: 10.1109/TVT.2015.2406877.
[83] M. Wazid et al., "Design of lightweight authentication and key agreement protocol for vehicular Ad hoc networks," *IEEE Access*, vol. 5, pp. 14966–14980, 2017, doi: 10.1109/ACCESS.2017.2723265.
[84] Y. Wang, H. Zhong, Y. Xu, J. Cui, and G. Wu, "Enhanced security identity-based privacy-preserving authentication scheme supporting revocation for VANETs," *IEEE Syst. J.*, vol. 14, no. 4, pp. 5373–5383, Dec. 2020.
[85] S. Tangade, S. S. Manvi, and P. Lorenz, "Trust management scheme based on hybrid cryptography for secure communications in VANETs," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5232–5243, May 2020, doi: 10.1109/TVT.2020.2981127.
[86] L. Zhang, X. Meng, K.-K. Raymond Choo, Y. Zhang, and F. Dai, "Privacy-preserving cloud establishment and data dissemination scheme for vehicular cloud," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 3, pp. 634–647, Jun. 2020, doi: 10.1109/TDSC.2018.2797190.

Gianpiero Costantino received the M.Sc. degree in 2007 and the Ph.D. degree in 2011. He is currently a Researcher with the Italian National Research Council (CNR). He is also working for the Trust, Security and Privacy Group, Institute of Informatics and Telematics, Pisa. He is the coauthor of about 50 scientific articles. His research interests include security and privacy on intelligent transportation system (ITS), cyber security and safety on automotive, and privacy and security platform for multi-modal transport to support privacy-preserving and secure analytics to users.



Marco De Vincenzi received the M.Sc. degree in data science and business informatics from the University of Pisa in 2020. He has been in the Automotive Industry for the past six years with a focus on quality management systems like ISO 9001:2015, IATF 16949:2016, and data analysis. He has participated in European projects like NGI-Trust COSCA and E-Corridor, respectively, for privacy policy analysis and information security. His research interests include data privacy, cyber security in intelligent transportation system (ITS), and the analysis of standards like ISO/SAE 21434 and UNECE WP29 R155.

Fabio Martinelli received the M.Sc. degree from the University of Pisa, Pisa, Italy, in 1994, and the Ph.D. degree from the University of Siena, Siena, Italy, in 1999. He is currently a Research Director of the Consiglio Nazionale delle Ricerche (CNR), Pisa, where he leads the Cyber Security Project area. He is involved in several steering committees of international WGs/conferences and workshops. He manages research and development projects on information and communication security. His main research interests include security and privacy in distributed and mobile systems and foundations of security and trust.

Ilaria Matteucci received the M.Sc. degree in 2003 and the Ph.D. degree in 2008. She is currently a Researcher of the Trust, Security and Privacy Group, Institute of Informatics and Telematics of CNR. Her main research interests include formal methods for secure systems, analysis of data sharing and policies on personal data privacy, automotive defensive and offensive cyber security, with particular reference to security properties of the CAN-bus protocol and possible vulnerabilities of in-vehicle network, and ITS security issues. She participates in National and European Projects in the field of information security.

Open Access funding provided by 'Consiglio Nazionale delle Ricerche-CARI-CARE' within the CRUI CARE Agreement