Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2014, Article ID 120626, 15 pages
http://dx.doi.org/10.1155/2014/120626

Research Article

Applying Information Hiding in VANETs to Covertly Report Misbehaving Vehicles

Jose Maria de Fuentes, Jorge Blasco, Ana Isabel González-Tablas, and Lorena González-Manzano

Computer Security Lab (COSEC), University Carlos III of Madrid, Avenida de la Universidad, 30 Leganes, 28911 Madrid, Spain

Correspondence should be addressed to Jose Maria de Fuentes; jfuentes@inf.uc3m.es

Received 4 November 2013; Accepted 20 December 2013; Published 5 February 2014

Academic Editor: Deyun Gao

Vehicular ad hoc networks (VANETs) are a new communication scenario in which vehicles take an active part. Real-time reporting of misbehaving vehicles by surrounding ones is enabled by in-vehicle sensors and VANETs. Thus, sensors allow detecting the misbehavior whereas VANETs allow sending the report to the authority. Nevertheless, these reports should pass unnoticed by the reported driver to avoid his/her potential reprisals. Information hiding techniques could be used to allow vehicles to transmit information covertly. In this work, two mechanisms for vehicle reporting are proposed based on two information hiding techniques—*subliminal channels* and *steganography*. The approach is to embed information into beacon messages either in the signature process (subliminal channel) or altering the least significant bits of selected sensorial fields (steganography). Results show that the proposal is computationally feasible for current vehicular devices and that it is possible to configure the system to operate in highways, secondary roads, and urban maps.

1. Introduction

Vehicular ad hoc networks (VANETs) are a new communication context in which vehicles can exchange information. They form a successful internet-of-things scenario that will be applied in the short term. VANETs are one of the enabling technologies of intelligent transportation systems (ITSs).

An interesting ITS application is the automatic reporting of misbehaving drivers by other drivers (or their vehicles). To be part of the VANET, vehicles need a communication and processing device called on-board unit (OBU). This device exchanges data with nearby vehicles mainly for traffic safety purposes. These data are mostly obtained from current vehicle-mounted sensors. Therefore, these technologies enable vehicles to automatically perceive the behavior of near ones. In fact, they have already been applied to use nearby vehicles as witnesses to defend against unfair punishments [1].

The capacity to monitor surrounding vehicles' behavior, along with the immediacy of VANETs, could significantly shorten the reporting process [2]. However, the reporting message could be observed by the misbehaving driver. This is because of the shared nature of VANETs. If the report is known by the offender, he/she could take reprisals against the reporting vehicle. Thus, its content must be concealed.

Encrypting the report would make it unreadable for the reported driver, since it would be encrypted using a key only known by the reporter and the authority. However, it must be noted that most VANET messages are related to traffic safety and thus they are sent in the clear. Therefore, an encrypted message being sent short after the illegal action would raise reasonable suspicions on the misbehaving vehicle. Even if it will never be sure on the actual message content (i.e., he/she will not know if it is actually a report or if it is referred to him/her), these suspicions would potentially be enough to take reprisals against the reporting vehicle. Given the terrible consequences that such an action may have, it would be useful to have a mechanism that could conceal the *message existence* (and not only its contents) from the reported vehicle.

Information hiding techniques allow sending data promoting that it passes unnoticed to undesired receivers.

Particularly, *subliminal channels* and *steganography* are two representative mechanisms [3]. A subliminal channel hides messages in the way an algorithm is applied over the normal-looking communication. On the other hand, steganography hides the secret by modifying some parts of the transmitted message. It must be noted that, thanks to information hiding techniques, it is not necessary to introduce a new message to convey data; it is sent embedded in regular messages and only when it is needed (e.g., a misbehaving action has been detected).

Sensor data is of outmost relevance in VANETs. They enable having real-time information on the traffic status. The use of information hiding techniques has been previously explored over sensorial data. It has been applied for proving ownership or integrity of sensor generated data [4–6]. However, to the best of our knowledge, the use of these information hiding techniques in VANETs has not been explored yet.

The goal of this work is to introduce information hiding techniques, particularly subliminal channels and steganography, to enable vehicles to send reports about misbehaving ones. Particularly, the *beacon* message will be taken as the carrier. This is a well-known VANET-related message that contains the sender current status in terms of position, speed, heading, and so forth. According to current standards, they are signed and sent every 100 ms to 1-hop VANET entities [7].

The approach of this work is to hide the complaint in the way the signature is calculated (subliminal channel) or within the beacon's sensorial data fields (steganography).

Subliminal channels can be used in this context since ECDSA the signature algorithm in IEEE 1609.2 standard for security in VANETs is not subliminal-free [8]. Steganography may be used in these fields because sensorial measurements are subject to some inaccuracy—there are some unrepresentative bits that may be used to embed data.

Different pros and cons may be found for both techniques. Subliminal channels are interesting because they do not alter beacon information. However, high-capacity subliminal channels in ECDSA require the sender and receiver to share an authentication key [9]. This issue must not be suitable for privacy-careful drivers which do not fully trust the Authority (i.e., the report receiver). Concerning steganography, the situation is exactly inverse. Whereas such a key sharing is not necessary, it involves modifying some bits within a beacon. Furthermore, future potential enhancements over sensors may decrease their inaccuracy, thus limiting the steganographic capacity.

Taking into account these issues, both mechanisms are adopted as alternatives in the proposed approach. This promotes the validity of the proposal even if any of the aforementioned limitations affects one mechanism. The capacity, robustness, and feasibility of the proposed approaches are evaluated. Results show that they are feasible for current vehicular devices and that at least one configuration setting exists in which they are operational for common scenarios (highways, secondary roads, and urban environments).

Paper Organization. Section 2 provides a brief background on information hiding techniques. Section 3 describes the considered model. Section 4 describes the proposed technique based on the subliminal channel, whereas Section 5 introduces the one which relies on steganography. Section 6 focuses on how these mechanisms may be applied in a real-world setting. Section 7 evaluates the security and feasibility of both techniques. Finally, Section 8 concludes the paper.

2. Information Hiding Techniques

In this section, the main information hiding techniques are described. A systematic revision of these techniques was performed by Petitcolas et al. [10]. In their paper, they identify four main families of techniques: steganography, subliminal channels, anonymity, and copyright marking. Neither anonymity nor copyright marking are interesting for our paper since their application is rather different from our goal. Therefore, only steganography and subliminal channels are relevant techniques.

Section 2.1 focuses on subliminal channels. Afterwards, a particular type of subliminal channel is studied due to its relationship in the proposal. Particularly, subliminal channels for the ECDSA algorithm (which is the digital signature mechanism for vehicular environments, according to the related security standard IEEE 1609.2 [11]) are presented in Section 2.2. Finally, Section 2.3 briefly presents the foundations of steganography.

2.1. Subliminal Channels. A subliminal channel can be defined as any communication link that hides messages in elements that were not originally intended for communication. Under this general definition, Zander et al. identified two main techniques to build these channels [12].

(1) *Timing Channel.* The time in which an action is performed has an intrinsic meaning. For example, if a message is sent in an odd second (e.g. 27th second within a minute), it would represent a bit value (say "0"); whereas if it is sent in an even one (e.g., 12th second) it would represent the opposite value (say "1").

(2) *Storage Channel.* The use/absence of an element represents a value. For example, if a message content is bigger than a predefined threshold, it would represent "1" and "0" otherwise.

Apart from the previous categories, the way in which an algorithm or protocol is applied may also be used to build a subliminal channel [9]. For example, selecting a given value for an algorithm parameter may be interpreted in a particular way by the receiver. Therefore, it may be employed to communicate between both parties. For the purpose of this work, a particular kind of this category is further explained in Section 2.2.

2.2. ECDSA and Its Subliminal Channels. In this section, a short description of ECDSA is first presented. Afterwards, the concept of subliminal channel and the ones applying to the aforementioned algorithm are introduced.

2.2.1. ECDSA. ECDSA is a public key signature algorithm that is based on a finite field F_p, an elliptic curve $E(F_p)$ over F_p where q_{ec} pertains to $E(F_p)$ with q_{ec} prime, and a point $G \in E(F_p)$ of order q_{ec} [13].

In this algorithm, the private key is a random $d \in \{1, \ldots, q_{ec} - 1\}$, whereas the public key is pub $= d \cdot G$.

Using the private key, the signature over a certain message m is a pair (r, s) such that $r = f(k \cdot G) \bmod q_{ec}$, being $k \in \{1, \ldots, q_{ec} - 1\}$, randomly chosen and s is calculated from

$$s = k^{-1} \cdot (d \cdot r + h(m)) \bmod q_{ec}. \quad (1)$$

In the former expressions, $h(\cdot)$ represents a collision-free hash function, and $f(\cdot)$ is a function that transforms the x-coordinate of a point on $E(F_p)$ to an integer.

2.2.2. Subliminal Channels. Subliminal channels enable sending a secret by the way an algorithm is used. Generally speaking, the secret is inserted within one of these algorithm parameters or results [9].

There are two types of subliminal channels, namely, *broadband* and *narrowband* channels. The difference between both types is the amount of data that can be embedded—it is maximum in the broadband channel, whereas it is reduced in the narrowband one [14].

For the specific case of ECDSA, one broadband and three narrowband channels have been identified [9]. Even if the cited work is focused on DSA, at least the broadband subliminal channel is also valid for ECDSA [15]. In all of them, it is necessary to share some information between sender and receiver. For the broadband channel, the shared secret is the sender's private key. In the narrowband ones, only a prime number, a binary sequence, or a particular value for a given parameter is required.

One important issue in broadband channels is that if the warden knows exactly the content of the subliminal message transmitted, he can retrieve the private key of the sender. Thus, in order to prevent this issue, it is necessary to send the subliminal message encrypted.

2.3. Steganography. Steganography is the science that focuses on how to hide the existence of messages [16]. Steganography shall not be confused with cryptography whose main aim is to conceal the content of the message so only allowed parties are able to read it. On the contrary, steganography aims to hide the message itself. The first informal description of steganography was given by Simmons as the prisoners problem [3]. Simmons described two prisoners (Alice and Bob) who want to plot an escape plan. They must communicate through a warden (Willie) who will analyse any communication between them. If Willie ever suspects that Alice and Bob are exchanging secret information he will isolate them.

In order to achieve their goal, they should hide their messages into innocuous-looking ones (called *covers*), so Willie will not be aware of the real meaning of those messages. As a difference with subliminal channels, the use of steganography involves modifying the cover.

The main goal of steganography is to build embedding functions that enable inserting practical amounts of data into covers while being undetectable for an attacker [17]. To achieve this goal, there should not be statistical differences between the set of all possible covers and the set of covers that hide the secret (*stego-objects*). Thus, it should not be possible to detect whether an object has embedded information or not without the knowledge of the key.

According to Petitcolas et al., two types of steganography are identified, namely, linguistic steganography and technical one [10].

(i) *Linguistic steganography* involves transforming the message to conceal into a textual (i.e., natural language) representation. For example, given a secret bit stream, this procedure builds a (potentially random) text with some semantical meaning. The receiver performs the reverse operation on the text to retrieve the secret bit stream.

(ii) *Technical steganography* may be based on two underlying procedures. First, a grammar may be used to generate a well-formed (potentially random) carrier message that internally contains the secret. The undetectability of this mechanism relies on the degree of realism of the created carriers. Second, parts of a carrier message may be altered to hide the secret. To promote undetectability, performed changes must not alter the original message meaning. Therefore, any source of redundant data is preferred for this purpose.

3. System Model

In this section, the model considered in this work is presented. First, the participant entities are introduced in Section 3.1. Section 3.2 introduces the system requirements. The threat model is described in Section 3.3. In Section 3.4, the selected message to hide the misbehavior report is described. Afterwards, the secret message structure is presented in Section 3.5, and the working assumptions are described in Section 3.6.

3.1. Participant Entities. In the proposed model, there are seven entities at stake (see Figure 1). The reporting vehicle (through its on-board unit, OBU) is the entity that sends the report related to a purported offending vehicle. This report is sent to in-range road-side units (RSUs). To enable the communication between OBUs and RSUs, a vehicular ad hoc network (VANET) is established. RSUs are common VANET-related static entities placed aside the roads that connect OBUs to service providers and the authority. For this particular context, RSUs send the received reports to the decision support system (DSS), which is managed by the Authority. DSS reveals the embedded data and sends it to the inspector who evaluates the relevance of the report and, if necessary, sends it to the report manager to proceed with the enforcement process.

In order for DSS to perform its operations, it interacts with the certification Authority (CA). CA manages the lifecycle of vehicular pseudonym-based short-lived certificates.

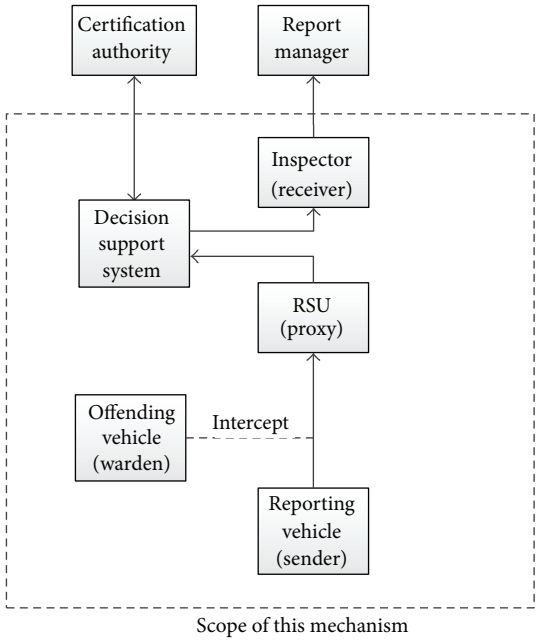

Figure 1: Model entities.

Vehicles are equipped with sensors that measure the vehicle's status (position, speed, heading, etc.). Sensorial data are assumed to be firstly sent to the event data recorder (EDR) device [18].

3.2. System Requirements. The envisioned system has to fulfil the following four main requirements:

(i) *Undetectability.* Secret information must remain undetectable for unauthorized parties.

(ii) *Reduced Computational Workload for RSUs.* Roadside units have to minimize their computational workload. Particularly, they cannot perform the message decryption or retrieval, and they must be able to determine, autonomously, whether a received message may contain secret information or not.

(iii) *Reduced Computational Workload for DSS.* DSS must only process messages in which it is plausible for them to contain a secret.

(iv) *Robustness.* The proposed approaches must contain countermeasures against the incidental data loss produced within the vehicular network.

3.3. Threat Model. Four entities are fully trusted, namely, report manager, CA, inspector, and DSS. All of them are related to the Authority or government in force, so they are under its physical and logical control. Furthermore, their interconnecting networks are also assumed to be fully reliable—no chance to access or manipulate the exchanged information.

Concerning RSU, it may be compromised by a malicious attacker to eavesdrop all exchanged messages to and from DSS. Even if the attacker could deactivate this device, this threat is left out of the scope as it requires physical countermeasures to be fully addressed.

With respect to the vehicular entities, the offending vehicle (i.e., the warden) cannot block the message sent, but, only eavesdrop it. The communication network may also be eavesdropped, and it is subject to incidental message losses. Denial-of-service attacks are assumed to have been countermeasured.

3.4. Message Structure to Hide the Report. The selected message to hide the secret (i.e., the misbehavior report) is the *beacon* message. According to standard SAE J2735, it contains several sensorial data fields (e.g., position, speed, heading, etc.) describing the sender's current status. It is sent every 100 ms. to nearby nodes (1 km away at most), and it is not routed [7].

There are two main reasons to perform this selection. On one hand, it is periodically sent by all vehicles, which enables an almost continuous communication channel to hide information. On the other hand, sensorial information is subject to errors caused by the limited accuracy of sensors. These errors lead to a set of unrepresentative bits that may be altered without causing a relevant threat to the data reliability. This is a beneficial situation for steganographic mechanisms.

One interesting issue is that according to SAE J2735, beacons are sequentially numbered [7]. This fact helps in relating different fragments of a given secret.

3.5. Secret Message. The misbehavior report to be secretly transmitted contains the following three fields (Figure 2).

(i) *Misbehaving action* (4 bits): it will identify the type of the reported misbehaving action.

(ii) *Message payload* (32 bits): it will be filled with the misbehaving vehicle identifier. Although this is a temporal pseudonym, it is the only publicly known identifier available to the reporting vehicle.

(iii) *Random section* (2 bits): it contains meaningless data. It is only included in the subliminal-based approach to promote that the secret may be sent using this technique.

The size of the random section is motivated by the probability for a message not to be transferable through a subliminal channel. According to Simmons, the total amount of messages that cannot be sent is $|e/1 - e| \approx 1.58$ messages. Having 2 bits of random section enables having 4 message structures for the same secret, thus enabling sending all desired secrets [14].

3.6. Working Assumptions. The proposed mechanisms are intended to work under the following five assumptions. First, vehicles' sensors are compliant with IEEE 1616 [18] and SAE J2735 [7] resolution and accuracy, and their inaccuracies (or errors) are random.

Second, each beacon is signed by the sending vehicle and the corresponding public key certificate is sent along with the beacon. This assumption is in line with recent mandates on

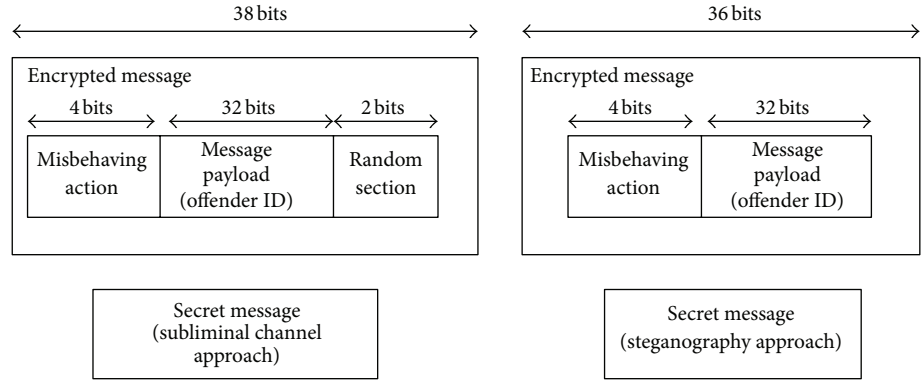

Figure 2: Structure of the reporting message to hide into beacons under both approaches.

vehicular security as stated in IEEE 1609.2 [11]. Related to this point, the third assumption is that vehicles will be using the same pseudonym while a single secret is sent.

Fourthly, concerning the subliminal-based approach, the vehicular cryptomaterial (i.e., public/private keypairs) is generated by the certification authority. This is one of the certificate request models identified in IEEE 1609.2 [11].

Finally, each vehicle is equipped with a set of (at least) 14 passwords. Each password is the result of encrypting for CA (using CA's public key) one permutation of the vehicle identification number (VIN). Each password may be used in a predefined set of seconds within each minute. According to ISO 3780, VIN has 17 alphanumeric elements [19]. For the sake of simplicity, we will assume that all elements are transliterated and represented by 4 bits. Thus, the whole VIN (and thus, each password) is 68 bits long.

It is also assumed that the sender will perform each sending operation for a single report using a different password. The selected amount of passwords is based on the quantity of repetitions required to promote that the message arrives taking into account data losses in VANETs. This issue is analysed in Section 7.1.

4. Subliminal Channel Architecture

In this section, the approach based on subliminal channels is described. At first, all subliminal channel techniques introduced in Section 2.1 are available. However, in the approach taken in this paper, the beacon message is selected as the carrier for the secret (see Section 3.4). Based on this decision, neither timing nor storage channels could be practical. Mandated by standards, beacons are regularly sent at periodic intervals, so the sender cannot choose when to send it. On the other hand, their structure is also well-known, so it is not possible to alter it at will. Even if they have one optional part (part II, as defined in standard SAE J2735 [7]), it is not intended to be frequent. Therefore, its mere existence could contribute to reveal the existence of the subliminal channel.

Considering these issues, only ECDSA-related subliminal channels (see Section 2.2) are available. Taking into account that broadband channels offer the maximum capacity, they are selected herein to maximize the efficiency. As explained in Section 2.2, it is possible because the vehicular cryptomaterial is known by the certification Authority, so the vehicle (sender) private key is already known by this entity.

Channel capacity is introduced in Section 4.1, whereas the embedding and revealing functions are presented in Sections 4.2 and 4.3, respectively.

4.1. Channel Capacity. According to Simmons, the channel capacity for a broadband channel is given by $l - e$, where $e < 10^{-47}$ and $l = \lceil \log_2(q_{ec}) \rceil$ [14]. Following the recommendations contained in standards for efficient cryptography (SEC), the suggested value for q_{ec} is 128 bits long [20]. Thus, $l = \lceil \log_2(q_{ec}) \rceil = 128$. In this situation, the channel capacity is in practice 127 bits, which is greater than the subliminal message size (38 bits, including the random section). Thus, there is no need to fragment the secret.

4.2. Embedding Function. In order for the vehicle to subliminally send the secret, it is taken as the value k of the ECDSA algorithm (Figure 3). Given that the warden may know the exact content of the secret, he could recover the reporting vehicle signing key (recall Section 2.2). Thus, the secret is encrypted before being used as the subliminal message. For this purpose, Simmons' method (which relies on Vernam encryption) is applied [14]. The key to be used in this step is the VIN-based password suitable for the time mark of the beacon at stake. As this password is 68 bits long (recall Section 3.6), which is greater than the message to encrypt, it may be used as the key for the process. It must be noted that using the VIN as key would also be possible, but having it encrypted limits the attacker's probability of success.

The signature process over the beacon Beacon(B_{ID}) is performed. As a result, the values r and s are obtained as usual (recall Section 2.2.1). Before proceeding with the submission, the sender checks that (2) holds:

$$h\left(\mathrm{Beacon}\left(B_{\mathrm{ID}}\right)\right) + d \cdot r \neq 0. \tag{2}$$

Equation (2) ensures that the secret message may be retrieved by the receiver. Otherwise, the secret message is altered in its random content section unless the previous condition is satisfied. The so-formed signed beacon is sent to DSS through the RSU. Given the amount of beacons that may be received

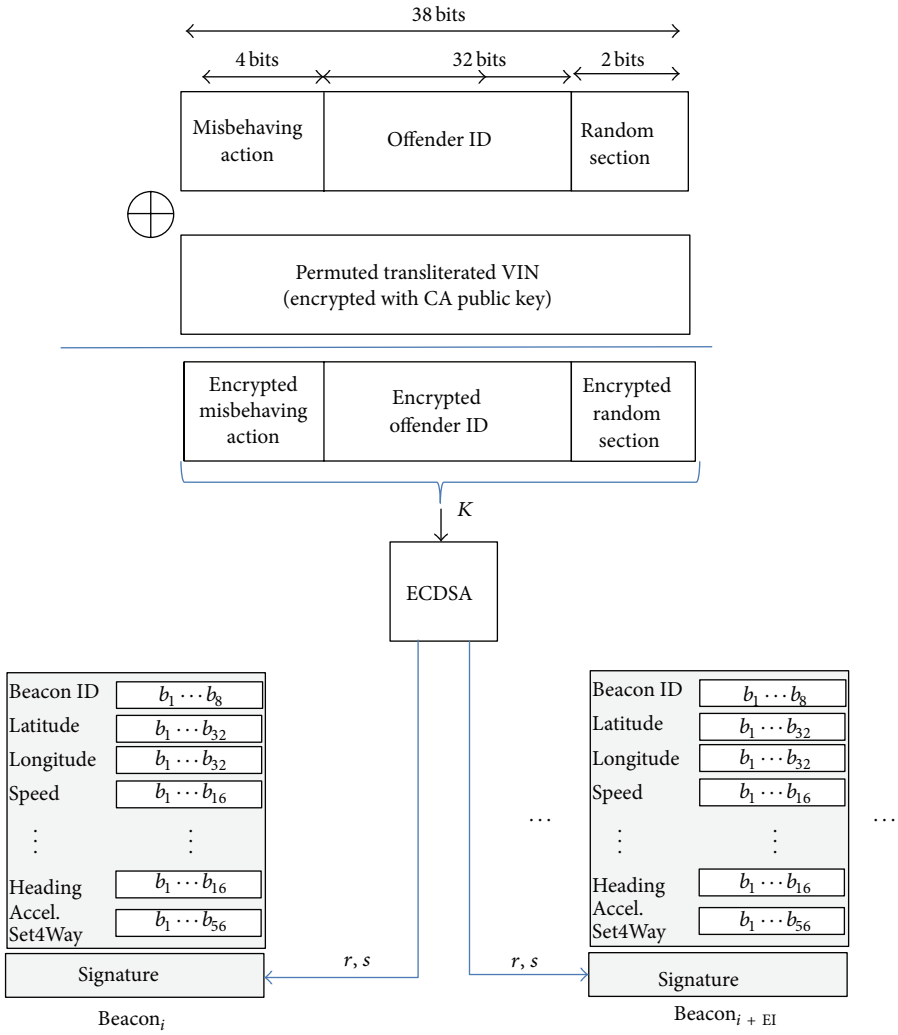

Figure 3: Embedding function for the approach based on subliminal channels.

by every RSU, a system parameter embedding interval (EI) is introduced. Particularly, only beacons whose identifier is a multiple of EI may contain subliminal information. In this way, only those beacons are sent to DSS for potential evaluation. This decision reduces the workload on this entity.

As the beacon is sent from the vehicle using the vehicular communication channel, the message may get lost. Thus, to promote that the message is received by DSS, it is re-sent several times. The amount of repetitions to be made for a report is given by the system parameter R. The analysis on the effect of this parameter in the global robustness of the system is discussed in Section 7.1.

It may happen that even if the beacon ID is a multiple of EI, the beacon does not contain any kind of report. Whereas this involves a waste of resources for RSU/DSS, the selection of a Vernam cipher (which is extremely fast) alleviates the computational workload.

4.2.1. Preventing False Positives. One critical issue to ensure the success of this approach is to prevent false positives, that is, beacons that contain a well-formed hidden message that was not intentionally inserted by the sender. To address this issue, the sender must avoid using misleading values of k when no hidden data is inserted. These values are those that lead to a valid secret message structure.

4.3. Revealing Function. The revealing function is applied periodically to the stream of received beacons (see Figure 4). This forces DSS to temporarily store these beacons (Section 7.3 analyses the amount of storage required). The process starts by the latest received ones, provided that their signature is successfully verified. Thus, for every received beacon Beacon(B_{ID}) (which are only those whose identifier is a multiple of EI), (3) is applied to reveal the encrypted secret message m'. Consider

$$m' = s^{-1} \cdot \left(h\left(\mathrm{Beacon}\left(B_{\mathrm{ID}}\right)\right) + d \cdot r\right) \bmod q. \tag{3}$$

To perform this operation, DSS retrieves the private key of the beacon's sender (i.e., d) from the Certification Authority (CA).

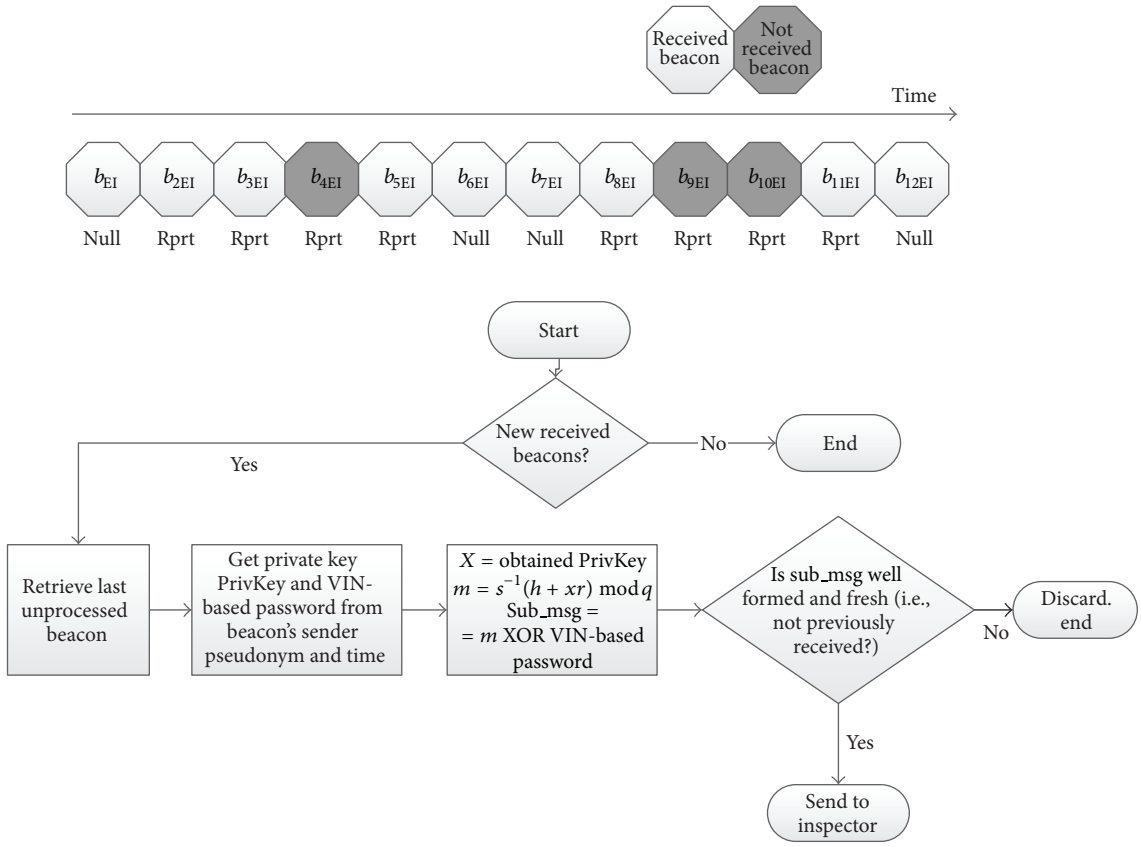

Figure 4: Subliminal approach. Revealing function workflow.

In order to decrypt m', DSS gathers the corresponding password, that is, the encrypted permutation of the transliterated VIN suitable for the beacon's time mark (recall Section 3.6). This password is applied to m' using a Vernam cipher leading to the subliminal message in the clear, referred to as sub_msg.

Once sub_msg has been obtained, it is checked whether it contains a valid misbehaving action code and if it has not been previously received (recall that every report will be sent R times, so it may be repeatedly received). If so, it is sent to inspector for evaluation and further action. If it was previously received, it is discarded. If it was not well-formed, the message may be discarded. However, in this last case the revealing procedure of the steganography-based technique (see Section 5.3) may be launched. This decision depends on how the system is applied in the real-world scenario. This issue is addressed in Section 6.

5. Steganography Based Architecture

In this section, the approach based on steganography is described. The first issue to address is to choose a particular type of steganography (see Section 2.3). Recalling that the beacon message is selected as the carrier for the secret, building a linguistic-based steganographic mechanism is not feasible since it is not a natural language element. Similarly, the first technical steganography procedure cannot be used herein. In such a procedure, the secret would be transformed into a beacon. The reverse transformation would enable retrieving the secret. However, it must be noted that beacons contain sensorial information which is far from random. Sensor measurements must be realistic to be credible. For example, current position or time values must be as close to their actual values as possible.

Considering these facts, modifying parts of the beacon is the best technical steganography procedure available for our purposes. The rest of this section describes the core of this technique. The capacity of each beacon is analysed in Section 5.1. Afterwards, the embedding and revealing functions are described in Sections 5.2 and 5.3, respectively.

5.1. Cover Message Capacity Analysis. In this work, it is considered to be acceptable to alter the sensorial value v_{measured} to a value v_{stego} that is within the range determined by the sensor's accuracy accy; that is, $v_{\text{stego}} \in [v_{\text{measured}} - \text{accy}, v_{\text{measured}} + \text{accy}]$.

The capacity of one data element d_i, that is, the number of values that can be encoded in certain sensorial data element, will be given by the ratio between the accuracy accy of the element and its resolution res plus one (to take into account the value provided by the sensor), as stated in

$$\text{capacity}_{d_i}\,(\text{bits}) = \left\lfloor \log_2 \left(\frac{\text{accy}_{d_i}}{\text{res}_{d_i}} + 1 \right) \right\rfloor. \quad (4)$$

The sensorial data is obtained first by the motor vehicle event data recorder (subject to IEEE 1616 [18]), and then the beacon message is constructed according to the SAE J2735 standard [7]. To calculate the capacity of each sensorial data element, the accuracy and resolution defined in the aforementioned standards have been analysed. While the EDR standard establishes the required resolution and accuracy, J2735 only describes the resolution of each field. Thus, in the calculations, the accuracy described in the IEEE 1616 standard has been used.

Table 1 specifies the maximum capacity of each beacon sensor field and the whole capacity of the message, 13 bits, considering the minimum capacity provided by both standards. Using this lower value is the most conservative approach, as it enables embedding the data at any point in the process, that is, before or after the sensorial data has been recorded in the EDR or prepared to be transmitted within a beacon.

5.2. Embedding Function. The proposed embedding technique consists in replacing the least significant bits of the sensorial data elements with those of the secret message (Figure 5).

As the vehicular communication channel is subject to data losses, a simple repetition scheme is selected. In this way, as it happened in the subliminal channel technique, every report will be repeated a number R of times. Nevertheless, if the same secret were identified by the attacker it would raise suspicions. For this purpose, it is necessary to prepare the message in such a way that every repetition leads to a different embedded message.

The proposed preparation is analogous to the one applied for the subliminal approach. Essentially, the secret (in this case, without the random section) is encrypted with a password using a Vernam cipher. The password is again a particular permutation of the VIN encrypted for the CA. The choice of the permutation to apply depends on the time mark of the beacon.

Once the message is prepared, it is inserted into the selected sensorial fields of the beacon. As the length of the secret (36 bits, recall Section 3.5) is higher than the capacity of each beacon (13 bits, recall Section 5.1), it is necessary to fragment the secret. The total amount of fragments (and thus, required beacons) is referred to as nb_{msg} and it is calculated in

$$nb_{\text{msg}} = \left\lceil \frac{36\,\text{bits/msg}}{13\,\text{bits/beacon}} \right\rceil = 3\,\text{beacons/msg}. \quad (5)$$

The embedding function protects first the secret message, splits it, and then embeds the fragments on nb_{msg} beacon messages.

The last step is to send the R instances of the secret to the receiver. As it happened in the subliminal based approach, an embedding interval EI is used—the secret may only be embedded in beacons whose identifier B_{ID} is multiple of EI. Apart from alleviating the receiver's workload, it reduces the global error introduced into the sensorial information—in a given amount of beacons, the introduced error is lower.

5.2.1. Preventing False Positives. As it happened in the subliminal channel, it is critical to avoid a well-formed hidden message when it is not inserted by the sender on purpose. To prevent these false positives, the sender must keep an eye on the bits affected by this embedding function for all beacons whose identifier is multiple of EI. In case that they could lead to valid revealed hidden information when it is not intended, the sender must transform it into an invalid secret. Given that the Vernam cipher operates in a bit-by-bit basis, it is enough to use a reserved action code to represent the absence of data.

5.3. Revealing Function. To obtain back the secret message, the revealing function reverts the embedding operations, performing in reverse order the process shown in Figure 5. The receiver does not know in advance if a reporting message is embedded in a beacon. Thus, it must proceed as if every beacon, among those eligible (i.e., considering the EI value), could contain the beginning of the secret. This is done by appending the extracted bits to a bitstream and by using a decryption window that moves along. In order to relate all fragments, the receiver benefits from the assumption that all of them are sent under the same pseudonym. Furthermore, this condition does not hold among repetitions, so each one is sent under a different identity. If any of the beacons including a message fragment is lost during transmission, the receiver will discard the existing beacons under the same pseudonym and will restart the process with the next received set.

6. Practical Settings

The subliminal channel technique (Section 4) and the steganography-based one (Section 5) are two different alternatives to convey the secret message.

In order to select which technique to apply, this section describes the two different practical settings that are envisioned.

(i) *Preestablished Selection.* In this setting, the Authority and the vehicle owner establish in advance which technique to apply. Two different choices are available.

  (1) *Always Subliminal.* This setting is suitable for drivers who share their private key with the Authority. They will benefit from the fact that improvements on sensor technologies will not affect their ability to send reports.

  (2) *Always Steganography.* This is the preferred setting for drivers that do not share their private key with the Authority. They will be able to send reports even if future revisions of IEEE 1609.2 standard determine a subliminal-free version of the ECDSA algorithm.

(ii) *On-the-Fly Selection.* The technique to apply is chosen by the sending vehicle on the fly. This enables full flexibility at the sender's side, as it will have both mechanisms available. In this setting, the receiver will first evaluate whether the subliminal channel has been used and, if it is not the case, it will assess the

Table 1: Capacity of each beacon sensorial data field, maximum introduced error and overall capacity of beacon messages.

Considered sensorial fields	Ratio $\text{accy}_{1616}/\text{res}_{1616}$	Ratio $\text{accy}_{1616}/\text{res}_{\text{J2735}}$	Capacity IEEE 1616	Capacity J2735	Maximum error
Latitude	600	10	9 bits	**3 bits**	$0.018'$
Longitude	600	10	9 bits	**3 bits**	$0.018'$
Speed	50	13	5 bits	**3 bits**	0.216 km/h
Heading	10	80	3 bits	**3 bits**	—
X acceleration	0	9	**0 bits**	3 bits	—
Y acceleration	0	9	**0 bits**	3 bits	—
V acceleration	0	0	**0 bits**	0 bits	—
Yaw rate	1	10	**1 bit**	3 bits	$0.1^{\circ}/s$
Overall (independ.)			**27 bits**	**24 bits**	
Overall (combined)			**13 bits**		

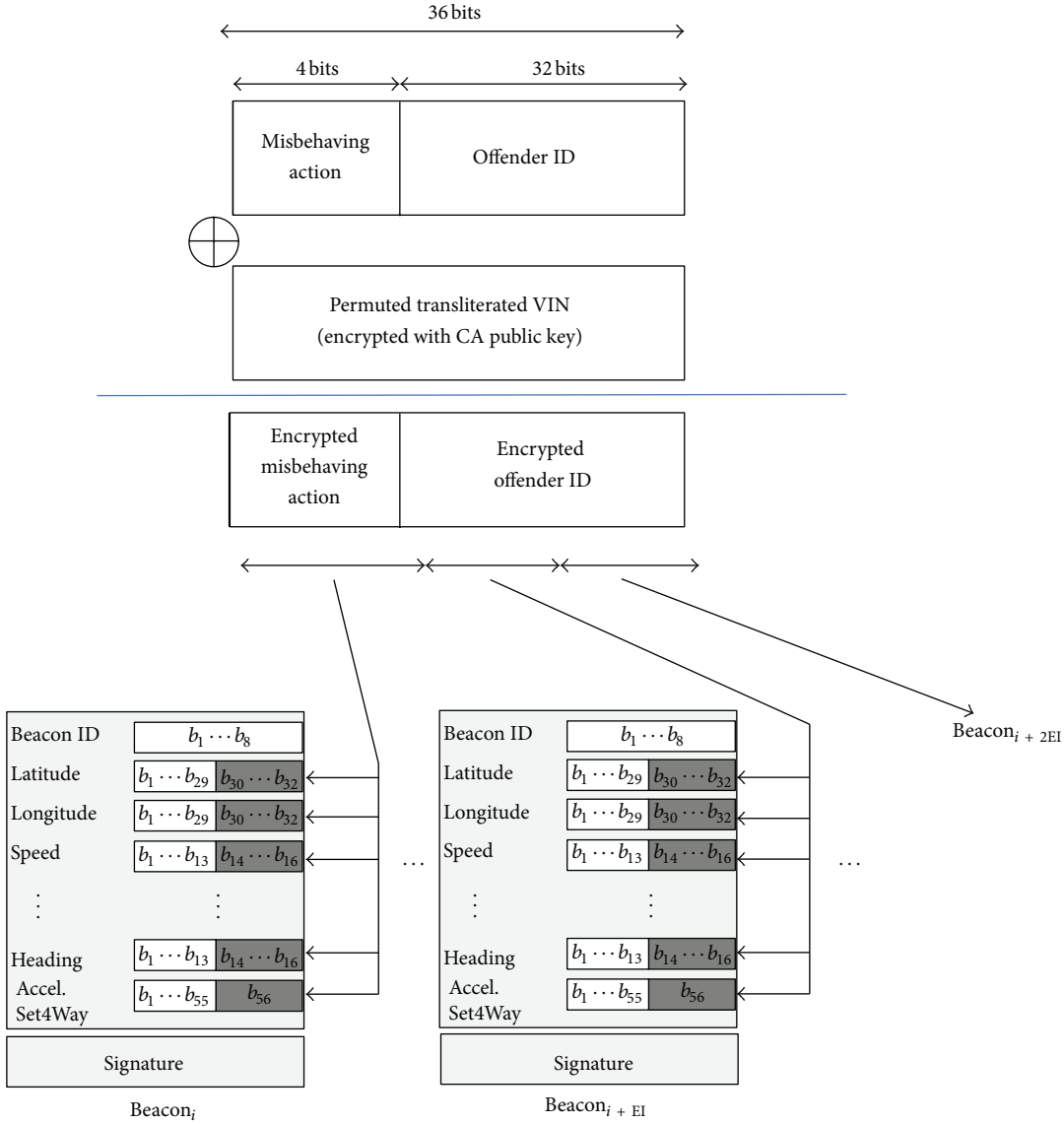

Figure 5: Embedding function for the steganographic approach.

use of steganography. This decision is motivated by the fact that the receiver is not aware of the type of information hiding technique that is in use. Therefore, it is necessary to execute both revealing procedures one after the other before determining that a given beacon does not contain hidden data. The resulting revealing workflow is shown in Figure 6. It is a slight modification of Figure 4.

7. Evaluation

In this section, the proposed system is evaluated. First, the system robustness given certain configuration is assessed in Section 7.1. The configuration is given by the system parameters EI (which determines the beacons that can contain hidden information) and R (which specifies the amount of times each report must be repeated to counter eventual data losses). The computational and operational feasibility of the system are discussed in Sections 7.2 and 7.3, respectively. Considering these results, the requirements analysis is presented in Section 7.4.

7.1. Robustness. The communication reliability of DSRC affects the robustness of the proposed system, as there is a nonnegligible probability of losing a sent packet. In this section, the minimum number of repetitions $R_{\min}$ under which the system is robust (to a certain probability $p_{\text{threshold}}$) is studied.

Let p_{beacon} and p_{msg} be the probability of successful reception of a beacon and the whole secret message, respectively. If the reception of each beacon is considered an independent event, p_{msg} can be calculated as a function of nb_{msg} (i.e., the amount of fragments in which the secret is divided) and p_{beacon} as

$$p_{\text{msg}} = \left(p_{\text{beacon}}\right)^{nb_{\text{msg}}}. \tag{6}$$

The success probability p_{success} is defined as the probability that at least one of the R repetitions has arrived. Thus, p_{success} can be calculated using (6), resulting in

$$p_{\text{success}} = 1 - \left(1 - p_{\text{msg}}\right)^{R}. \tag{7}$$

To ensure the system robustness, $p_{\text{success}} > p_{\text{threshold}}$ under all configurations of the system. This condition imposes the minimum number of repetitions (referred to as $R_{\min}(p_{\text{threshold}})$), which is graphically shown in Figure 7. For this calculation, p_{beacon} is assumed to be 0.58 which is the value estimated in [21] for the delivery ratio in VANETs for packets sent from 400 meters (vehicle-to-vehicle). Note that Figure 7 shows not only $R_{\min}$ for $nb_{\text{msg}} = 1$ (the case of the proposed subliminal channel system as there is no fragmentation) and $nb_{\text{msg}} = 3$ (the value of the proposed steganographic approach), but also $nb_{\text{msg}} = 9$, which is significantly higher than the proposed approaches. In this way, the effects of varying nb_{msg} are illustrated. Focusing on the proposed mechanisms, it may be seen that for a $p_{\text{threshold}} = 0.95$, only 4 repetitions are required for the subliminal approach, whereas the steganographic approach requires 14. As this is the amount of required passwords (recall Section 3.6), the system can reach this probability. For all values of $p_{\text{threshold}}$, the amount of repetitions is higher in the proposed steganographic approach than in the subliminal channel one.

7.2. Computational Feasibility. In this section, it is analysed if all participants are computationally capable of sending and receiving hidden messages. For the sake of clarity, the sender feasibility and the receiver one will be analysed separately.

In order to simplify the reasoning, it is assumed that both OBUs and RSUs operate in continuous access mode to the wave control channel (CCH) [22]. This channel is employed to transmit beacon messages. Thanks to this type of access, no channel switching overhead has to be considered and no synchronization is required for this task.

7.2.1. Sender. In both approaches, the first action is to prepare the secret message (T_{prep}). This task may involve a greater amount of time if a prepared message cannot be sent through the subliminal channel (recall Section 3.5). In this situation, the message *random section* field has to be altered.

Once the message is prepared, it is necessary to take it as an input for the signing process. For this purpose, it is encrypted using a Vernam cipher. Determining which key (i.e., which encrypted permutation of the VIN) has to be applied takes a time $T_{\text{genVnmKey}}$ to derive a fresh key. Afterwards, the cipher operation has to be applied. According to Simmons, this is performed using multiplication in Galois fields, which takes a time T_{multVnm} [14]. The result of this operation is taken as an input for the signature (subliminal approach) or fragmented and embedded into beacon fields (steganography approach). This fragmentation and insertion take a time T_{subst}. All these tasks are repeated for every message repetition (i.e., R times). Taking into account these issues, the time T_{SND} spent by the sender is calculated as shown in (8), where $\alpha = 1$ if the steganography approach is used and 0 otherwise:

$$T_{\text{SND}} = T_{\text{prep}} + R \cdot \left(\left(T_{\text{genVnmKey}} + T_{\text{multVnm}}\right) + \left(\alpha \cdot T_{\text{subst}}\right)\right). \tag{8}$$

In order to guarantee the system's computational feasibility, the time T_{SND} must be lower than the time to send the nb_{msg} beacons. This ensures that while a message repetition R_i is being sent, the next one R_{i+1} can be prepared. Otherwise, it could not be possible to send the next repetition right after the previous one.

In order to estimate the sender cost, it is assumed that the most computationally significant operations are the cryptographic ones, as the remaining operations are simple manipulations of messages. The multiplication in Galois field (T_{multVnm}) seems to be the most representative one, remarkably in embedded devices. According to [23], this multiplication takes 672,492 clock cycles to be performed in an embedded platform. Commercial OBUs such as Locomate (http://www.aradasystems.com/LocoMate-OBU/, last accessed in September 2013) have a 680 Mhz processor.

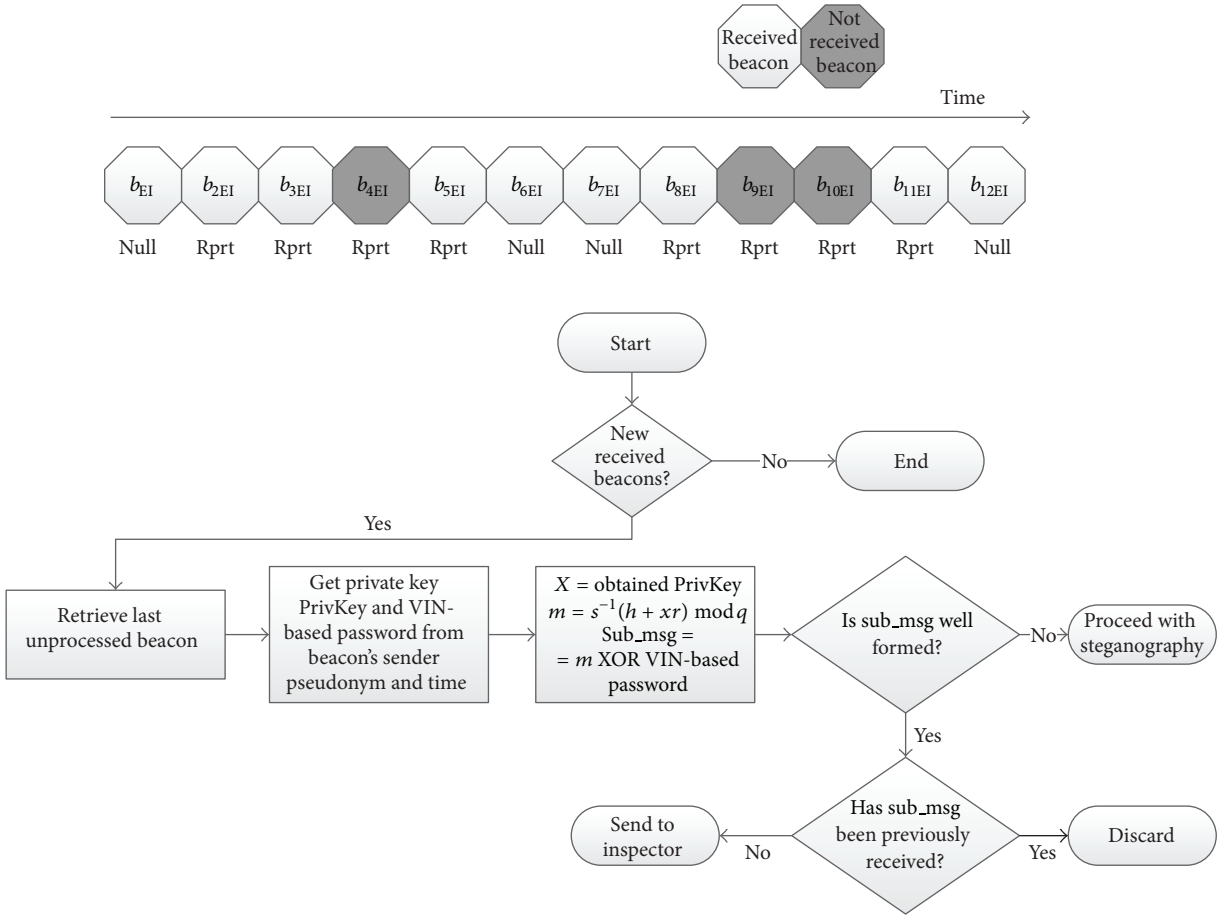

Figure 6: Subliminal approach. Revealing function workflow for on-the-fly selection setting.

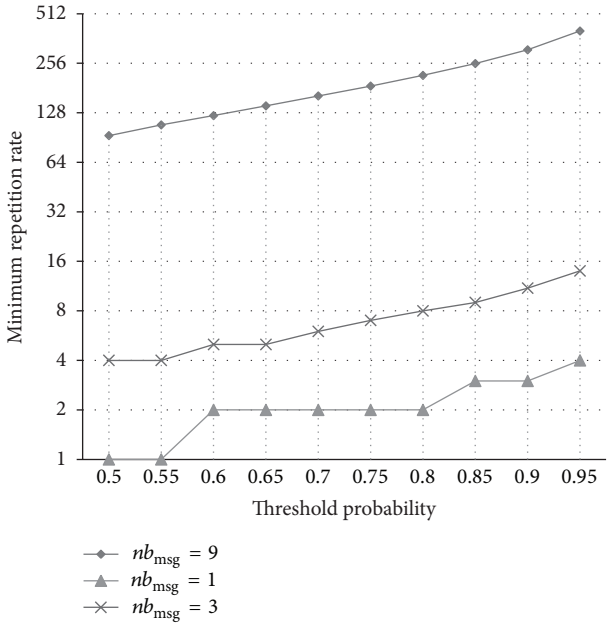

Figure 7: Analysis of the minimum repetition rate R_{min} once a threshold probability $p_{threshold}$ is selected. Note the logarithmic scale on the Y-axis.

Therefore, the aforementioned operation takes 0.98 ms in a vehicular device. This time is significantly smaller than the time to send the hidden message in the most stringent case. Using a subliminal channel (recall that no fragments exist) and EI = 1, all repetitions are continuously sent and they are separated only by the beaconing rate T_{beacon} = 100 ms according to current standards.

Even if the previous figures indicate that there is enough time to perform these operations, it must be noted that all beacons sent are signed. For this reason, the temporal overhead introduced by signature generation over sent beacons, T_{SG}, and signature verification of received beacons, T_{SV}, must be also considered. Performance figures for embedded platforms taken from [24] state that T_{SG} = 16.856 ms and T_{SV} = 45.381 ms. In the time period to send all nb_{msg} fragments of a given secret, the sender will spend a time T_{frag}, described in (9), where δ is the mean number of incoming signed beacons within a beacon period T_{beacon}:

$$T_{frag} = nb_{msg} \cdot \left(T_{SG} + \delta \cdot T_{SV}\right). \quad (9)$$

With these figures, it is obvious that a vehicle can hardly verify more than one incoming signed beacon from neighbouring vehicles. To overcome this limitation, we assume

that the periodic or context-adaptative verification strategies proposed in [25] are applied. Therefore, δ can be adjusted to assure that the vehicle copes with the overhead introduced by the regular cryptography-related beaconing tasks and leaves some time to embed the next message repetition. Equation (10) shows this condition, where $\overline{\delta}$ is the adjusted mean number of incoming secure beacons that will be actually verified within a T_{beacon} period and $T_{nb_{\text{msg}}\text{beacons}}$ is the time to send all fragments of a given secret message:

$$T_{\text{SND}} + nb_{\text{msg}} \cdot \left(T_{\text{SG}} + \overline{\delta} \cdot T_{\text{SV}}\right) \leq T_{nb_{\text{msg}}\text{beacons}}. \quad (10)$$

Under this assumption, it may be concluded that the proposed approach is computationally feasible for the sender.

7.2.2. Receiver. With respect to the receiver, there are two entities at stake—RSU and DSS. The RSU's task is to decide whether the beacon identifier is multiple of the parameter EI or not. The time to perform this operation is assumed to be negligible. Therefore, the time T_{RCV} is entirely spent by DSS and is composed of three operations. First, the encrypted subliminal message has to be extracted, either by solving the ECDSA-based equation (subliminal mechanism) or by retrieving the bits at stake from beacons (steganographic one). This operation requires a time $T_{\text{extHidMsg}}$ and is repeated as many times as fragments have the secret (i.e., nb_{msg} times). Second, the appropriate Vernam key has to be retrieved ($T_{\text{retVnmKey}}$) and applied to decrypt the subliminal message (T_{multVnm}). Finally, the received message is processed (T_{procMsg}) to determine if (1) it has a subliminal message, or (2) it contains steganographic information or (3) it does not contain hidden information (recall Figure 4). The latter is the worst case as all resources had been unnecessarily wasted. For this case, the revealing process of the subliminal message (T_{SUBL}) and the steganographic one (T_{STEG}) are sequentially applied. Therefore, the time taken by the receiver T_{RCV} is shown as

$$\begin{aligned} T_{\text{RCV}} &= T_{\text{SUBL}} + T_{\text{STEG}} \\ &= \left(T_{\text{extHidMsg}} + T_{\text{retVnmKey}} + T_{\text{multVnm}} + T_{\text{procMsg}}\right) \\ &\quad + \left(nb_{\text{msg}} \cdot T_{\text{extHidMsg}} + T_{\text{multVnm}} + T_{\text{procMsg}}\right). \end{aligned} \quad (11)$$

It is important to note that in the subliminal approach the secret is not fragmented (and therefore, only one extraction is applied) and that the key is the same for both approaches in a given beacon (so there is no need to retrieve it in the steganographic process).

Among all the described operations, there are three remarkable ones. First, solving the ECDSA-based equation in the subliminal approach ($T_{\text{extHidMsg}}$) may require a significant workload. The use of hardware-based acceleration (which is reasonable for these devices) would be advisable to ensure availability. Second, due to the cryptographic design, the decryption operation T_{multVnm} is the same as the one performed by the sender; that is, $T_{\text{multVnm}} = 0.98$ ms. In fact, this operation would be faster as DSS resources are greater than those from the in-vehicle platform. On the other hand, retrieving the necessary password from CA (i.e., $T_{\text{retVnmKey}}$) seems to be a significant time-consuming task as it requires a search operation within CA's database. To the best of the authors' knowledge, any performance figure that illustrates this cost does not exist.

In order to ensure that DSS can cope with all beacons, it must be able to timely process all of them. Thus, for a given second (12) must hold, where numVeh is the amount of vehicles whose beacons will be received by DSS:

$$\left(\frac{(10\ \text{beacons/sec.} \cdot \text{numVeh})}{\text{EI}}\right) \cdot 0.98\ \text{ms.} \leq 1000\ \text{ms.} \quad (12)$$

In the event that EI = 1, numVeh = 102 cars. This figure gives the maximum threshold for simultaneous vehicles in a given area to ensure that it is computationally feasible for the receiver.

7.3. Operational Feasibility. In the proposed approaches, it is assumed that the reporting vehicle sends the required amount of beacons while being in the range of a set of RSUs. Nevertheless, the amount of required RSUs connected to a single DSS has not been characterized. It mainly depends on the vehicle's speed. This section analyses the system feasibility regarding this issue. The suitability of different system configurations (parameters EI and R) for different types of roads according to the vehicle's speed and the distance travelled are also analysed. Afterwards, the storage needs for DSS are also studied. For the sake of brevity, only the steganographic-based approach is considered as it involves a greater amount of fragments per secret message (e.g., higher nb_{msg}). In this situation, the amount of beacons to send a set of hidden messages is higher, thus potentially involving a greater amount of RSUs.

7.3.1. Amount of Required RSUs: Suitability to Different Driving Scenarios. Assuming that a specific system setting is selected by choosing certain values of EI and R, the required total number of beacons used to transmit a report is $N = R \cdot nb_{\text{msg}} \cdot \text{EI}$. On the other hand, as beacons rate is br (beacon/s) = $1/T_{\text{beacon}}$, the number of beacons M_{RSU} that a vehicle can actually transmit to one RSU will depend on the communication range r between both and the relative speed v of one in respect to the other: $M_{\text{RSU}} = (r \cdot \text{br})/v$ (with v in m/s and r in m). If a set of ρ RSUs is considered, the number of beacons M increases accordingly: $M = \rho \cdot M_{\text{RSU}}$.

To guarantee the system's operational feasibility, M must necessarily be greater than N. By design, $r = 1000$ m, br = 10 beacon/s, and $nb_{\text{msg}} = 3$. Therefore, the operational feasibility comes determined by

$$R \cdot \text{EI} \cdot v \leq \frac{(\rho \cdot r \cdot \text{br})}{nb_{\text{msg}}} = \rho \cdot 3333,33. \quad (13)$$

We analyse the system's operational feasibility in nine scenarios specified by the vehicle's speed and the distance travelled (Figure 8). Considered speeds are those common in highways (120 km/h), secondary roads (80 km/h), and urban environments (40 km/h). In highways, vehicles are

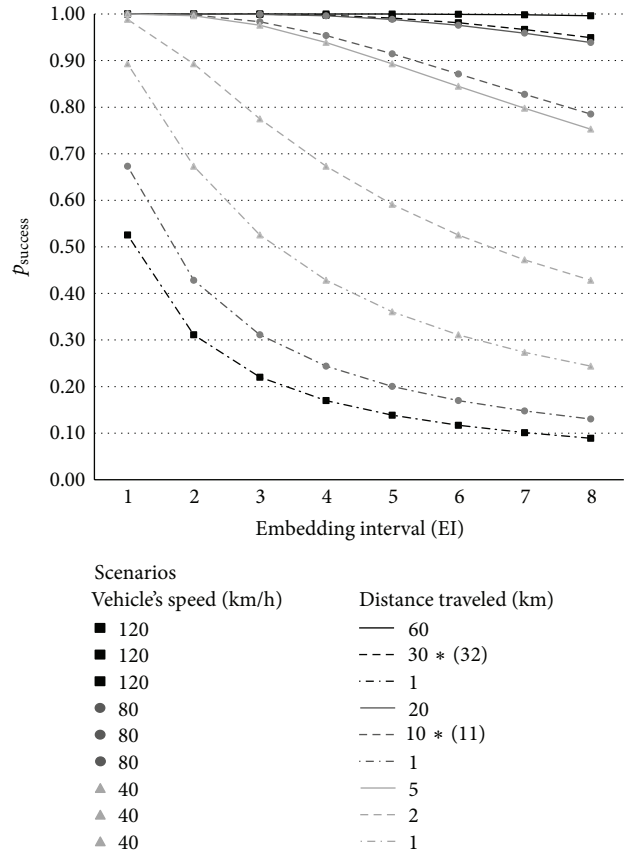

Figure 8: Success probability in the considered scenarios.

assumed to have mean trip lengths of 60 km, 30 km, and 1 km; in secondary roads, analysed mean trip lengths are 20 km, 10 km, and 1 km; finally, for urban roads; 5 km, 2 km, and 1 km are considered. It has been assumed that RSUs are placed every kilometre, so the number of travelled kilometres is equal to ρ.

Figure 8 shows the probability of success $p_{success}$ in these scenarios as a function of the embedding interval EI, which can be seen as a measure of the introduced error (the higher EI, the lower the error). Note that we have avoided using *exactly* some of the travelled distance values (30 km and 10 km) to increase the figure's readability.

From our point of view, the system is considered to be feasible if $p_{success} \geq 0.75$. Thus, the system is not feasible if only one RSU is available (distance travelled or $\rho = 1$), if driving at 40 km/h or 80 km/h, or if driving at 120 km/h and EI ≥ 3. If more than one RSU is considered, there is at least one feasible setting in each scenario. Generally speaking, $p_{success}$ lowers as EI rises, because a higher embedding interval gives less chance to embed data for the same amount of time. Therefore, with speeds of 120 km/h and 80 km/h, the system is feasible for all considered values of EI if travelled distance equals 60 km or 30 km. It also happens when speed equals 40 km/h and distance is 5 km, while when distance equals 2 km and 1 km only EI = 3 and EI = 1 satisfy the feasibility condition, respectively.

7.3.2. Storage Needs for DSS. The amount of beacons at stake must be stored by DSS. As the described revealing procedures are applied periodically to the set of received beacons, it is necessary to temporarily store them.

Speaking generally, the amount of beacons to store depends on six variables, (1) the beacons sent by one vehicle while being in range $\text{beacons}_{1\text{veh}}$, (2) the density of vehicles $\text{density}_{\text{veh}}$, (3) the probability of successful transmission p_{beacon}, (4) the amount of RSUs ρ (i.e., the driving path length), (5) the embedding interval EI, and (6) the beacon size ($\text{size}_{\text{beacon}}$). It may be seen that the amount of beacons sent by one vehicle also depends on its speed as driving slower enables sending more beacons while being in a given RSU's range. Taking these factors into account, the general expression for the storage of DSS is $\text{Store}_{\text{DSS}}$ (14). Consider

$$\begin{aligned}\text{Store}_{\text{DSS}} &= \frac{(\text{beacons}_{1\text{ veh}} \cdot \text{density}_{\text{veh}})}{\text{EI}} \\ &\quad \cdot p_{\text{beacon}} \cdot \rho \cdot \text{size}_{\text{beacon}} \\ &= \frac{(((r/\text{vehSpeed}) \cdot \text{br}) \cdot \text{density}_{\text{veh}})}{\text{EI}} \\ &\quad \cdot p_{\text{beacon}} \cdot \rho \cdot \text{size}_{\text{beacon}}.\end{aligned} \tag{14}$$

From the storage point of view, the worst situation is when the amount of RSUs is higher and the embedding interval is minimum. Thus, in order to give a worst-case value on this issue, it will be assumed that $\rho = 60$ and EI = 1. As this amount of RSUs corresponds to the highway scenario previously defined, vehicles will be assumed to be driving at 120 km/h. It must be noted that even if the amount of RSUs connected to a single DSS may be higher, the considered trip length determines the maximum amount of RSUs that a vehicle may encounter. With respect to the vehicular density, the value 320 veh./km^2 will be taken, according to the worst scenario considered in [26]. Even if it is beyond the computational threshold (recall Section 7.2.2), it illustrates the situation in which DSS processing capabilities are higher than those considered in this work. The beacon size ($\text{size}_{\text{beacon}}$) will be 39 bytes according to SAE J2735 standard, plus the size of a digital signature and the public key certificate (181 bytes) as mandated by IEEE 1609.2 standard. The remaining variables (i.e., r, br, and p_{beacon}) will take their usual values.

Based on the previous values, the amount of storage required by DSS is $\text{Store}_{\text{DSS}} = 734.97$ Mbytes. This amount of beacons is created in the time interval in which all vehicles complete the whole driving path, which takes 30 minutes. This time should be the maximum period in which the revealing function should be applied to the set of received beacons. Even if this amount of storage seems reasonable for the envisioned technological context of DSS, it should be noted that for the sake of immediacy, the goal should be to reveal these reports as immediately as possible. At the light of this fact, 30 minutes seems to be the highest value if immediacy is to be achieved. Thus, the calculated amount of storage is the highest value in a practical working scenario.

7.4. Requirements Analysis. In a nutshell, both approaches have fulfilled the imposed requirements. A detailed explanation for each one is given below.

(i) *Undetectability.* The subliminal approach fulfills this property as the key used to encrypt the secret is unknown to the warden, so it cannot determine the mere existence of the secret. On the other hand, the steganographic approach keeps the modified value within the accuracy boundaries, thus avoiding to perform a noticeable change over the beacon data. Furthermore, this technique does not introduce statistical differences under the working assumptions. It must be noted that even if the identity of the reporting vehicle is known to the misbehaving one, there is no risk of reprisals. The reported vehicle cannot detect that it is being reported. The report itself is hidden from that vehicle, thanks to the said properties of the selected information hiding techniques.

(ii) *Reduced Computational Workload for RSUs.* The workload on RSUs is reduced in both mechanisms to determine whether the received beacon's identifier is multiple of the parameter EI. If it is the case, the message is transferred to DSS, being discarded otherwise.

(iii) *Reduced Computational Workload for DSS.* Related to the previous point, the parameter EI also reduces the workload on DSS, as only a fraction of beacons are candidates to contain hidden information. Furthermore, the steganographic approach enables DSS to save resources by discarding some beacons if not all fragments of a given repetition are successfully received.

(iv) *Robustness.* Both techniques follow a repetition strategy, thus decreasing the probability of secret data loss.

8. Conclusions and Future Work

In this work, two information hiding approaches (subliminal channels and steganography) have been proposed to enable vehicles sending hidden reports about other misbehaving ones. For this purpose, the report is embedded in the signature process of VANET beacon messages (subliminal channel) or within the least significant bits of selected sensorial data fields within these messages (steganography). For both approaches, the secret message structure, the cover capacity, and the procedures to protect, embed, and extract the secret data have been presented. The proposal has been evaluated in terms of the degree of robustness against communication errors, the required computational effort, and its feasibility in representative scenarios. It has been shown that it is suitable to current vehicular computational devices and that it may be used (under different settings) for common values of vehicle's speed and distance travelled.

Considering the previous facts, the proposed system has four main advantages. Particularly, (1) it enables sending sensitive data (a traffic misbehaving report) over public, shared communication media (the VANET), *unnoticed* to all entities except for the intended receiver; (2) as reports are embedded in beacon messages, they can be sent immediately after the misbehaving action is detected; (3) the system is robust against incidental data losses typical of VANETs; and (4) it is computationally feasible for the vehicular environment, taking into account current state-of-the-art devices and realistic driving scenarios.

The approach taken in this paper serves as a starting point for the application of covert channels in vehicular networks. Given that several malicious uses could be given to this technique (e.g., driver-to-driver radar/police control warning), we believe that this will encourage the research community to invest efforts in this direction.

Future work on this area will have three main directions. First, we will evaluate other encryption techniques for the steganographic approach to improve its performance. An open issue is to increase the efficiency for both the sender (avoid the overload caused by repetitions) and the receiver (avoid processing beacons without a secret). Second, a practical evaluation with real sensors will be performed to contrast the degree of randomness of their errors. Third, the adoption of other information hiding mechanisms (e.g., timing subliminal channels) will be studied. For this purpose, other messages structures to hide the message must be taken into consideration.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is partially founded by Ministerio de Ciencia e Innovacion of Spain under Grant TIN2009-13461 (project E-SAVE). Authors want to thank Dr. M. I. González-Vasco and Dr. Julio C. Hernandez-Castro for their helpful suggestions. Furthermore, authors would like to thank the anonymous reviewers for their valuable comments.

References

[1] J. M. de Fuentes, L. Gonzalez-Manzano, A. I. Gonzalez-Tablas, and J. Blasco, "WEVAN—a mechanism for evidence creation and verification in VANETs," *Journal of Systems Architecture*, vol. 59, no. 10, part B, pp. 985–995, 2013.

[2] J. M. de Fuentes, A. I. Gonzalez-Tablas, J. Lopez, and A. Ribagorda, "Towards an automatic enforcement for speeding: enhanced model and ITS realization," *IET Intelligent Transport System*, vol. 6, no. 3, pp. 270–281, 2012.

[3] G. J. Simmons, "The prisoners' problem and the subliminal channel," in *Proceedings of CRYPTO '83: Advances in Cryptology*, pp. 51–67, 1984.

[4] J. Fang and M. Potkonjak, "Real-time watermarking techniques for sensor networks," in *Society of Photo-Optical Instrumentation Engineers*, vol. 5020 of *Proceedings of SPIE*, pp. 391–402.

[5] R. Sion, M. Atallah, and S. Prabhakar, "On watermarking numeric sets," in *Digital Watermarking*, vol. 2613 of *Lecture Notes in Computer Science*, pp. 130–146, Springer, 2003.

[6] H. Guo, Y. Li, and S. Jajodia, "Chaining watermarks for detecting malicious modifications to streaming data," *Information Sciences*, vol. 177, no. 1, pp. 281–298, 2007.

[7] SAE J2735, Dedicated Short Range Communications (DSRC) Message Set Dictionary, 2009.

[8] J.-M. Bohli, M. I. G. Vasco, and R. Steinwandt, "A subliminal-free variant of ECDSA," in *Information Hiding*, vol. 4437 of *Lecture notes in Computer Science*, pp. 375–387, 2007.

[9] R. Baecker, *Subliminal Channels in Cryptographic Systems*, Ruhr-Universitat Bochum, 2009.

[10] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding—a survey," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1062–1078, 1999.

[11] Institute of Electrical and Electronics Engineers (IEEE), "Trial-use standard for wireless access in vehicular environments—security services for applications and management messages," IEEE Std 1609.2 (rev D15), 2012.

[12] S. Zander, G. Armitage, and P. Branch, "Covert channels and countermeasures in computer network protocols," *IEEE Communications Magazine*, vol. 45, no. 12, pp. 136–142, 2007.

[13] National Institute of Standards and Technology (NIST), *Digital Signature Standard (DSS)*, 2009.

[14] G. J. Simmons, "Subliminal communication is easy using the DSA," in *Proceedings of EUROCRYPT 93: Advances in Cryptology*, pp. 218–233, 1993.

[15] L. Harn and G. Gong, "Elliptic curve digital signatures and accessories," in *Proceedings of the International Workshop on Cryptographic Techniques and E-Commerce*, pp. 126–131, 1999.

[16] N. F. Johnson and S. Jajodia, "Exploring steganography: seeing the unseen," *Computer*, vol. 31, no. 2, pp. 26–34, 1998.

[17] I. J. Cox, M. L. Miller, J. A. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, Morgan Kaufmann, 2008.

[18] Institute of Electrical and Electronics Engineers (IEEE), "Motor Vehicle Event Data Recorders (MVE-DRs)," IEEE Std 1616, 2005.

[19] International Standards Organization (ISO), *ISO 3780 Road Vehicles—World Manufacturer Identifier (WMI) Code*, 2009.

[20] Certicom Research, *Standards For Efficient Cryptography. Recommended Elliptic curve Domain Parameters*, 2000.

[21] F. Bai and H. Krishnan, "Reliability analysis of DSRC wire-less communication for vehicle safety applications," in *Proceedings of the Intelligent Transportation Systems Conference*, pp. 355–362, 2006.

[22] S. A. M. Ahmed, S. H. S. Ariffin, and N. Fisal, "Overview of Wireless Access in Vehicular Environment (WAVE) protocols and standards," *Indian Journal of Science and Technology*, vol. 6, no. 7, pp. 4994–5001, 2013.

[23] M. Khalil-Hani, A. Irwansyah, and Y. W. Hau, "A tightly coupled finite field arithmetic hardware in an FPGA-based embedded processor core for elliptic curve cryptography," in *Proceedings of the International Conference on Electric Design*, 2008.

[24] A. Festag, P. Papadimitratos, and T. Tielert, "Design and performance of secure geocast for vehicular communication," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 5, pp. 2456–2471, 2010.

[25] E. Schoch and F. Kargl, "On the efficiency of secure beaconing in VANETs," in *Proceedings of the 3rd ACM Conference on Wireless Network Security (WiSec '10)*, pp. 111–116, usa, March 2010.

[26] W. Viriyasitavat, O. K. Tonguz, and F. Bai, "Network connectivity of VANETs in Urban areas," in *Proceedings of the 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '09)*, June 2009.