network

MDPI

Article

A Low-Overhead Message Authentication and Secure Message Dissemination Scheme for VANETs

Hassan Mistareehi and D. Manivannan *

Department of Computer Science, University of Kentucky, Lexington, KY 40508, USA; hassan.mistareehi@uky.edu
* Correspondence: mani@cs.uky.edu; Tel.: +1-(859)-257-9234

Abstract: Given the enormous interest shown by customers as well as industry in autonomous vehicles, the concept of Internet of Vehicles (IoV) has evolved from Vehicular Ad hoc NETworks (VANETs). VANETs are likely to play an important role in Intelligent Transportation Systems (ITS). VANETs based on fixed infrastructures, called Road Side Units (*RSU*s), have been extensively studied. Efficient, authenticated message dissemination in VANETs is important for the timely delivery of authentic messages to vehicles in appropriate regions in the VANET. Many of the approaches proposed in the literature use *RSU*s to collect events (such as accidents, weather conditions, etc.) observed by vehicles in its region, authenticate them, and disseminate them to vehicles in appropriate regions. However, as the number of messages received by *RSU*s increases in the network, the computation and communication overhead for *RSU*s related to message authentication and dissemination also increases. We address this issue and propose a low-overhead message authentication and dissemination scheme in this paper. We compare the overhead, related to authentication and message dissemination, of our approach with an existing approach and also present an analysis of privacy and security implications of our approach.

Keywords: VANET; security and privacy; internet of vehicles; internet of things; intelligent transportation systems

Citation: Mistareehi, H.; Manivannan, D A Low-Overhead Message Authentication and Secure Message Dissemination Scheme for VANETs. *Network* **2022**, *2*, 139–152. https://doi.org/10.3390/network2010010

Academic Editor: Dongkyun Kim

Received: 22 December 2021
Accepted: 17 February 2022
Published: 7 March 2022

1. Introduction

Given the enormous interest shown by customers as well as industry in autonomous vehicles, the concept of an Internet of Vehicles (IoV) has evolved from Vehicular Ad hoc NETworks (VANETs). Thus, VANETs are likely to play an important role in Intelligent Transportation Systems (ITS). According to some estimates, the global market for IoV is likely to exceed USD 200 billion by 2024. Many auto manufacturers have programs in place for developing a platform for connecting to IoV services such as route management and smart parking. VANET consists of vehicles and *RSU*s. Each vehicle is equipped with On-Board Unit (OBU), which allows the vehicle to collect data from their environment, process, and send information to other vehicles and/or *RSU*s through wireless communication (e.g., Dedicated Short-Range Communication (DSRC)). Therefore, using Vehicle-to-Vehicle (V2V) communication, vehicles can send and receive alert messages. For example, modern vehicles have Emergency Electronic Brake Lights (EEBL). This system aims to warn other vehicles if there is a need for sudden hard braking, for example, in foggy weather, where visibility may become low and brake lights are not bright enough to be recognized by other drivers [1,2].

Vehicle-to-Infrastructure (V2I) communication can help with avoiding accidents. The *RSU* can collect and process the information from vehicles moving within its transmission range; looking at the data that had been analyzed, if an accident is about to happen, *RSU* broadcasts a warning message to vehicles in its transmission range so they can take appropriate action to avoid it [2,3]. A dynamic traffic congestion pricing system for IoV [4] has been proposed. In this system, to alleviate traffic congestion, the participating vehicles are rewarded for taking an alternative path. The proposed system is implemented using VANETs, which eliminate the need for installing a costly electronic toll collection system.

The authors in [5] proposed an accident prediction system for VANET. The crash risk in their system can be observed using velocity, driver fatigue, weather conditions, vehicles density, and crash location. They used a hidden Markov model to model the correlation between these observations and the crash risk. The results of their proposed system show the ability to detect potential crashes [5]. Over the past few years, researchers in both academia and industry have continuously worked on designing efficient schemes for privacy-preserving authentication and secure message dissemination in VANETs.

Clustering techniques have been used in V2V communication-based VANET architectures, wherein the network is divided into multiple clusters and one node in each cluster is selected as their Cluster Head (CH). The CH is responsible for all local cluster communication. This clustering technique helps with reducing the message overhead because it restricts the communication between CH and the members in its cluster. The CH can collect and also process and aggregate information from its cluster members and then propagate them to other clusters through other CHs [6,7]. Many researches proposed schemes [8,9] for electing CHs in each cluster based on specific parameters, such as vehicle location, vehicle speed, etc. Dividing the network into multiple clusters reduces the communication overhead and improves the network efficiency.

In infrastructure-based architectures for VANETs, vehicles use Road Side Units (RSUs) to form a VANET. In some schemes [10,11], vehicles authenticate each other, while in other schemes [12,13], vehicles use RSUs for authenticating disseminating messages sent by vehicles in its region. If traffic becomes heavy, it may not be possible for RSUs to receive messages about events observed by all vehicles in its region, authenticate them, and disseminate them in a timely manner, especially because the same event will be observed and sent by many vehicles in its region. In this paper, we address this problem and propose a solution.

In our approach, when the density of vehicles in an RSU's region is high, the RSU divides its region within its transmission range into several sub-regions and selects one vehicle in each sub-region as the Group Leader (GL). The GL selected in a sub-region is supposed to collect messages sent by vehicles in its sub-region, authenticate them, aggregate them, and forward them to the RSU. This reduces the overhead related to message authentication for the RSU.

Following are the major contributions of our work:

- We propose a low overhead message authentication and secure message dissemination scheme for VANETs. Vehicles themselves do not authenticate messages. RSUs are responsible for collecting, aggregating, authenticating and disseminating messages to vehicles.
- To reduce the message authentication overhead, RSUs can select some vehicles in its region as group leaders (GLs) to collect/aggregate messages from vehicles in their subregions and send them to the RSU for further aggregation and dissemination.
- Our scheme ensures authenticity and integrity of messages using digital signature based on public key cryptography.

The rest of the paper is organized as follows. We discuss some related works in Section 2. In Section 3, we describe our proposed approach. In Section 4, we present the security and privacy analysis of our approach. Finally, Section 5 concludes the paper.

Next, we discuss some related works.

2. Related Works

Cluster-based vehicular cloud architectures have been proposed in [14,15] for infrastructureless VANETs; under these approaches, vehicles are grouped into clusters based on their location, speed, computation capability, etc. Vehicles belonging to a cluster elect a Cluster Head (CH). The CH performs the creation, maintenance, and deletion of vehicles in that cluster. A scheme in [16] proposed a similar approach, where vehicles in a specific region form a vehicular cloud elect a broker among them. The broker collects the desired data from the vehicles and then sends it to a cloud server if further processing is required. Security-related issues are not addressed in these schemes. The authors in [15] designed a

secure communication protocol for exchanging messages among vehicles in a smart city using an Elliptic Curve Cryptography (ECC) technique. In their scheme, Cluster Heads (CHs) are responsible for communicating and verifying messages within their clusters, and the CHs are verified by the Certification Authority (CA). In this scheme, frequent CH elections could occur if vehicles move fast.

Many privacy-preserving authentication schemes, such as anonymous authentication [17], cooperative authentication [10], and dual authentication [18] have been proposed. For example, Azees et al. [17] proposed a PKI-based efficient anonymous authentication scheme with a conditional privacy-preserving (EAAP) scheme for VANETs. The vehicles and RSUs communicate anonymously to provide privacy and anonymity during the authentication process, and the TA can revoke a misbehaving vehicle and find out its real identity in case of dispute. This scheme is secured against different attacks (e.g., impersonation attacks, message modification attacks, etc). However, in the above schemes [10,17,18], vehicles communicate not only with each other but also with the RSUs to verify the authenticity of the messages.

Schemes presented in [19–21] used RSUs for authenticating, processing, and disseminating messages received from vehicles in its region. In [19], a safety warning system in fog-cloud-based VANETs using a Certificateless Aggregation Signcryption Scheme (CASS) have been proposed. Vehicles send traffic messages to the RSUs, which act as fog nodes. These fog nodes process and aggregate the received messages. These schemes [19–21] address the security and privacy issues for VANETs. However, they do not consider heavy densities of vehicles, which may cause increased computation and communication overhead.

In our scheme, vehicles do not form clusters among themselves. Each RSU can decide when and where to form clusters in its region, based on the density of vehicles and other parameters such as the region from which the RSU receives a large number of messages. In addition, the RSU assigns the Group Leader GL (the Group Leader is not elected) for each cluster, and the GL is responsible for collecting, authenticating, and aggregating the messages received from its cluster/group and for forwarding them to the RSU. The RSU is responsible for collecting the messages sent by the GLs in its region, authenticating them, aggregating them, and forwarding them to the vehicles in its region and/or other RSUs for further dissemination. This approach reduces the computation and communication overhead for the RSUs.

3. Proposed Approach

In this section, we present our system model and describe the proposed method for authenticated message dissemination in detail. The acronyms used in this paper are listed in Table 1.

Table 1. Notations.

Notation	Description
ID_A	Identity of Entity A
PID_A	Pseudo Identity of Entity A
M	A Message
v	Vehicle v
ts	Timestamp
PR_A	Private Key of Entity A
PU_A	Public Key of Entity A
K	Symmetric Key established between two communicating parties
$SIG_A(M)$	Signature of M Signed using A's Private Key
$H()$	Hash Function

Table 1. *Cont.*

Notation	Description
$E(M, K)$	Encryption of M with Key K
RSU	Roadside unit
GL	Group Leader
DMV	Department of Motor Vehicles
$Cert_v$	Certificate issued to vehicle v by the DMV
$Cert_{RSU}$	Certificate issued to RSU by the DMV

3.1. System Model

The system model for our scheme is shown in Figure 1. It consists of Department of Motor Vehicles (DMV), Road Side Units (RSUs), On-Board Units (OBUs), and Group Leaders (GLs). We describe the functions of these entities next.

- **DMV:** We assume that all vehicles are registered with a trusted authority (TA), such as the Department of Motor Vehicles (DMV), that administers the registration of the vehicles. The DMV is assumed to be trusted and cannot be compromised. The DMV generates its public and private keys (PU_{DMV}, PR_{DMV}) and distributes a PU_{DMV} to all RSUs and vehicles securely. In addition, the DMV generates pseudo-IDs (PID_v) for each vehicle, certificates corresponding to each pseudo-ID of a vehicle ($Cert_v$) where $Cert_v = E((PID_v, PU_v, ts), PR_{DMV})$, and certificates of RSUs ($Cert_{RSU}$) where $Cert_{RSU} = E((ID_{RSU}, PU_{RSU}, ts), PR_{DMV})$.
- **Vehicle:** Each vehicle is assumed to be equipped with an On-Board Unit (OBU) for computation and communication with RSUs as well as with other vehicles. The OBU stores the vehicle's public and private key pair (PU_v, PR_v), its pseudo-IDs (PID_v), its certificates corresponding to each pseudo-ID of the vehicle ($Cert_v$ signed by the DMV), and the public key of the DMV (PU_{DMV}).
- **RSU:** The Road Side Units (RSUs) are fixed entities along the roadside which facilitate V2V and V2I communication. RSUs are connected to each other and to the DMV, possibly through the Internet. In our scheme, a RSU collects the messages sent by the vehicles in its region, authenticates the messages, aggregates the messages, and forwards them to vehicles within its region, as well as to vehicles in other regions as needed.
- **Group Leader (GL):** Each RSU divides its region into sub-regions based on the density of vehicles in the region. Then, the RSU selects one vehicle in each sub-region as a GL. The GL is responsible for collecting, authenticating, and aggregating messages sent by vehicles in its sub-region and for sending them to the RSU. The GL is also responsible for receiving messages from the RSU, authenticating them, and disseminating them to vehicles in its sub-region.

  We describe the proposed method in detail next.

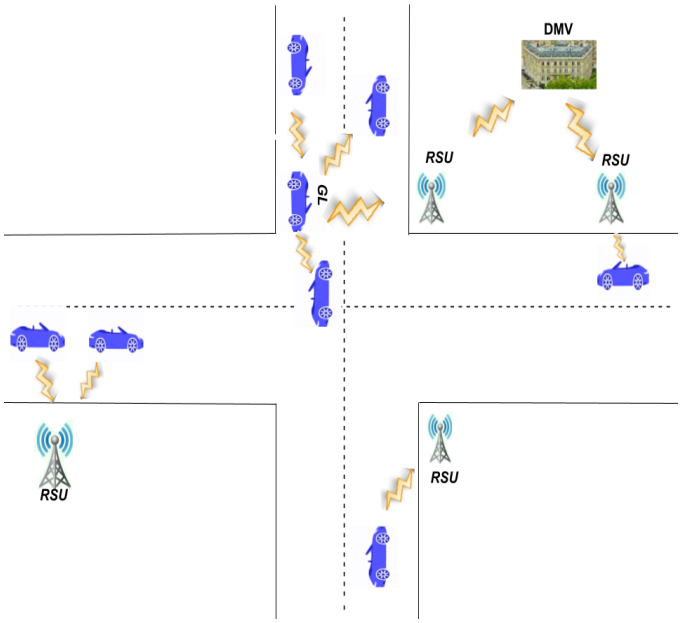

Figure 1. System model for VANETs.

3.2. Proposed Method

In our scheme, *RSU*s are responsible for verifying the authenticity and integrity of messages sent by vehicles before disseminating them to other vehicles or *RSU*s. If traffic is heavy in the region of an *RSU*, the *RSU* may not be able to receive messages from all vehicles in its region, process them, and disseminate them in a timely manner due to the authentication, aggregation, and communication overhead involved. To help *RSU* minimize this overhead, the *RSU* divides its region into sub-regions and selects one vehicle in each sub-region as the Group Leader (*GL*). These Group Leaders help the *RSU* with receiving, authenticating, and aggregating messages from vehicles in its sub-regions and forwards them to the *RSU*. The *RSU*, in turn, is responsible for collecting, authenticating, and further aggregating the messages received from all the *GL*s in its region, and for disseminating them to all vehicles in its region through the *GL*s or to vehicles in other regions through other *RSU*s, as necessary. Thus, *RSU*s incur less computation and communication overhead for collecting, authenticating, and disseminating messages. Following is the list of assumptions made in this paper:

1. We assume that the clocks of *RSU*s, the *DMV*, and the vehicles are loosely synchronized. This can be achieved using time received from a GPS. Messages are time-stamped using the local clock time to verify the freshness of the messages;
2. Certificates issued by the *DMV* for the vehicles and *RSU* are used for the authentication of vehicles and *RSU*s;
3. We do not address the issue of determining malicious vehicles or *RSU*s. Several approaches have been proposed in the literature to identify malicious entities in VANETs. Any of those approaches can be used for determining malicious vehicles. Once a vehicle is determined to be malicious, the *DMV* revokes its certificate and includes the certificate in the Certificate Revocation List (*CRL*). The *DMV* broadcasts the *CRL* to all *RSU*s when it changes. The *RSU*s, in turn, broadcast the *CRL* to vehicles in its region;
4. When a vehicle v enters the region of an *RSU* (i.e., v is within the transmission range of an *RSU*), even though v will be able to receive messages sent by the *RSU*, v may not be able to send messages directly to the *RSU* because the *RSU* may not be within the transmission range of v. In this case v uses an underlying routing algorithm to send messages to the *RSU* through other vehicles. Any of the many routing algorithms proposed in the literature can be used for that purpose.

Next, we describe our approach in detail.

When a vehicle v enters the region of an *RSU*: Each RSU periodically broadcasts its $Cert_{RSU}$. When a vehicle v enters an area covered by an RSU, v retrieves the public key of the RSU from $Cert_{RSU}$ and checks its CRL to see if this RSU's certificate has been revoked (the certificate of an RSU could be revoked if it is removed from the system). If not, then v sends a join request message M to the RSU. The join request message M contains its currently used PID_v, the corresponding certificate $Cert_v$, and a timestamp (ts). After receiving this message, the RSU checks the freshness of the message using the ts. Then, the RSU retrieves the public key PU_v and pseudo-ID PID_v of the vehicle from $Cert_v$, and checks the CRL to determine if the vehicle's certificate has been revoked. If not, then the RSU sends an accept message to v. The accept message contains a symmetric key K to be used for secure communication between the RSU and v, and a timestamp ts, encrypted using the public key PU_v of v; it also attaches the certificate of the RSU, signed by the DMV ($Cert_{RSU}$), and the signature of the RSU (SIG_{RSU}) to the message as follows:

$$M_1 = (RSU, PID_v, (E(\text{“}Accept\text{”}, K, ts), PU_v), Cert_{RSU}, SIG_{RSU}),$$

where

$$SIG_{RSU} = E(H(\text{“}Accept\text{”}, K, ts), PR_{RSU}).$$

Upon receiving the above accept message from the RSU, the vehicle uses the received ts to verify the freshness of the accept message. After that, it verifies the $Cert_{RSU}$ and the signature of the RSU. Algorithm 1 contains the algorithm illustrating the joining process of a vehicle v when v enters the region of an RSU.

Algorithm 1: When a vehicle v enters the region covered by an RSU

`When a vehicle` v `enters the region covered by an` RSU`:`
- Verifies $Cert_{RSU}$ received in the broadcasted message using PU_{DMV};
- Retrieves PU_{RSU} from the $Cert_{RSU}$;
- Computes $M_1 = (\text{“}Join\text{”}, ts)$;
- Encrypts M_1 using public key PU_{RSU} of RSU;
- Sends $M_1' = (PID_v, RSU, E(M_1, PU_{RSU}), Cert_v, SIG_v)$ to the RSU, where $SIG_v = E(H(M_1), PR_v)$

`When the` RSU `receives` M_1' `from` v`:`
- Decrypts M_1' using PR_{RSU};
- Verifies $Cert_v$ using PU_{DMV};
- Retrieves PU_v from $Cert_v$;
- Verifies the signature using PU_v;
- If verification succeeds {
  - Computes $M_2 = (\text{“}Accept\text{”}, K, ts)$;
  - // M_2 contains the acceptance message
  - // for the joining message from v;
  - // K is the symmetric key to be used between v and RSU;
  - Encrypts M_2 using public key PU_v of v;
  - Sends $M_2' = (RSU, PID_v, E(M_2, PU_v), Cert_{RSU}, SIG_{RSU})$ to v, where $SIG_{RSU} = E(H(M_2), PR_{RSU})$};
- Else { Discards $M2$; }

`When a vehicle` v `receives` M_2' `from` RSU`:`
- Decrypts M_2' using its private key PR_v to obtain M_2;
- Verifies SIG_{RSU} using PU_{RSU};
- If verification succeeds {
  - Stores (M_2);}
- Else { Discards $M2$. }

Next, we describe how an *RSU* selects Group Leaders in its region and informs them about being selected.

Informing selected vehicles as Group Leaders: When a vehicle v enters the region covered by an *RSU*, it sends a join message to the *RSU* after authenticating the *RSU*. Then, the *RSU* authenticates v and sends an "Accept" message, which includes a symmetric key K to be used between v and the *RSU*. Afterwards, the vehicle can send messages about sensed events to the *RSU*, encrypting them using K. If the *RSU* is not within the vehicle's transmission range, the messages are sent to the *RSU* using an underlying routing algorithm, as we mentioned earlier. Upon receiving "join" messages from vehicles in its region, an *RSU* can determine the number of vehicles in its region and their location. If the density of vehicles in the region of an *RSU* is low, the *RSU* does not need to select a *GL*. If the density of vehicles in an *RSU*'s region is high, it divides its region into sub-regions and selects one vehicle from each sub-region as the Group Leader (*GL*). After selecting *GL*s, the *RSU* informs the selected vehicles (*GL*s) of their leadership and sends a proof-of-leadership message $M_1 = E((\text{"}Leader\text{"}, PU_{GL_i}, ts), PR_{RSU})$. The *RSU* encrypts the M_1 using a symmetric key K, established between v and *RSU* when v entered the *RSU*'s region, attaches its signature (SIG_{RSU}) to the message, and sends the M_1', where $M_1' = (RSU, PID_v, E(M_1, K), SIG_{RSU})$, and $SIG_{RSU} = E(H(M_1), PR_{RSU})$.

When a *GL* receives the above message M_1' from the *RSU*, it decrypts the message using a symmetric key K and uses the received ts to verify the freshness of the message. After that, it verifies the signature of the *RSU* and stores M_1 as proof of leadership, so it can present it to the vehicles in its sub-region as proof that it is a leader. Algorithm 2 illustrates how an *RSU* informs the selected vehicles of their leadership (*GL*s). The *GL*s are responsible for authenticating, aggregating, and forwarding messages collected from vehicles in its sub-region. Thus, the *RSU* only needs to authenticate and process messages that come from *GL*s. Therefore, the communication and computation overhead for *RSU*s will be reduced. Moreover, when an *RSU* needs to send some message to all vehicles in its region or only to vehicles in some sub-regions, it will send that message only to the *GL*s in those sub-regions, which, in turn, will send it to all the vehicles in its sub-region.

Algorithm 2: Assigning Group Leaders (*GL*s) for selected vehicles by *RSU*

RSU `determines the number of vehicles and their locations in its region:`
  Based on the density of vehicles in the *RSU*'s region,
  If Density is high {
    RSU selects a set of vehicles as Group Leaders (*GL*s);
    For each vehicle selected as a *GL* {
      Computes $M_1 = (E(\text{"}Leader\text{"}, PU_{GL}, ts), PR_{RSU})$;
      Encrypts M_1 using symmetric key K;
      // K is the symmetric key established between v and
      // the *RSU* when v joined *RSU*'s region;
      $M_1' = (RSU, PID_v, E(M_1, K), SIG_{RSU})$,
      where $SIG_{RSU} = E(H(M_1), PR_{RSU})$;
      Sends M_1' to *GL*; } }
  `else`{
    No *GL*s are selected;
    RSU authenticates and process messages from all vehicles; }

`When a` *GL* `receives` M_1' `from` *RSU*:
  Decrypts M_1' using K;
  Verifies the signature using PU_{RSU};
  If verification succeeds{
    Stores (M_1) as proof of leadership;}
  Else {Discards M_1.}

Next, we describe how a vehicle in a sub-region establishes a connection with its Group Leader and communicates with its Group Leader.

When a vehicle v enters the sub-region of a GL: Each GL periodically broadcasts its public key PU_{GL} and the proof of leadership received from the RSU, namely, $E((\text{“}Leader\text{”}, PU_{GL}, ts), PR_{RSU})$. When a vehicle v enters a sub-region covered by a GL, it retrieves PU_{GL} from the proof of leadership. Then, v sends a join request message M to the GL; M contains a PID_v, $Cert_v$, and timestamp (ts). Upon receiving M, the GL checks the freshness of the message using ts. Then, the GL retrieves the PID_v and public key PU_v of the vehicle from $Cert_v$ and checks the CRL to determine if the vehicle's certificate has been revoked. After verification, GL sends an acceptance message and a symmetric key K to be used for secure communication between the vehicle v and the GL. The acceptance message M_1' contains the certificate of the GL, signed by the DMV ($Cert_{GL}$), a K, and a ts, encrypted using the public key PU_v of v as follows: $M_1' = GL, PID_v, (E(\text{“}Accept\text{”}, K, ts), PU_v), Cert_{GL}, SIG_{GL})$

Upon receiving the above acceptance message from the GL, v uses the received ts to verify the freshness of the message. After that, it verifies the signatures of the DMV and GL. Note that if v does not recieve proof of leadership from a GL (this happens when the RSU has not determined leaders due to low density of vehicles in its region), after entering an RSU's region, v sends/receives messages to/from the RSU directly, using an underlying routing protocol. Algorithm 3 illustrates the joining process when v is in the sub-region of a GL.

Algorithm 3: When vehicle v enters a sub-region covered by a Group Leader GL

```
When v enters the region covered by a GL:
  Receives proof of leadership message
  E(("Leader", PU_GL, ts), PR_RSU) from the GL;
  Retrieves PU_GL from the encrypted message using PU_RSU;
  Computes M_1 = ("Join", ts);
  Encrypts M_1 using public key of Group Leader PU_GL
  Sends M_1' = (PID_v, GL, E(M_1, PU_GL), Cert_v, SIG_v) to
  GL, where SIG_v = E(H(M_1), PR_v);

When a GL receives M_1' from v:
  Decrypts M_1' using PR_GL
  Verifies Cert_v using PU_DMV;
  Verifies the signature using PU_v;
  If verification succeeds{
    Computes M_2 = ("Accept", K, ts);
    // M_2 contains the acceptance of GL for v;
    // K is a symmetric key between v and GL for further
    // communication;
    Encrypts M_2 using public key PU_v of v ;
    Sends M_2' = (GL, PID_v, E(M_2, PU_v), SIG_GL) to v,
    where SIG_GL = E(H(M_2), PR_GL)};
  Else { Discards M_2;}

When v receives M_2' from the GL:
  Decrypts M_2' to obtain M_2;
  Verifies SIG_GL using PU_GL;
  If verification succeeds{
    Stores (M_2); }
  Else { Discards M_2; }
```

When a vehicle v wants to send a message M to its GL: When v wants to send a message M about an observed event to its GL, it signs and encrypts M and sends M_1 to

the GL, where $M_1 = (PID_v, GL, E((M, ts), K), SIG_v)$; here, ts is the timestamp, K is the symmetric key established between v and GL, and PID_v is the pseudo-ID of v.

When GL receives M_1, it decrypts the message using the symmetric key K and checks the freshness of the message using the ts. It uses a signature SIG_v to verify the authenticity and integrity of the message. Then, the GL aggregates the received message with the messages received from other vehicles in its sub-region and forwards the aggregated message to the RSU, and the RSU can further aggregate messages received from other GLs in its region and disseminate them to the appropriate sub-regions of its region or regions covered by other RSUs. Algorithm 4 shows this message collection and dissemination process.

Algorithm 4: Vehicle v sending a Message M to its Group Leader GL

```
When a vehicle v wants to send a message M about an observed event:
    Computes M1 = (PIDv, GL, E((M, ts), K), SIGv);
    Sends M1 to GL;
    // K is the symmetric key established in the
    // Algorithm 3.
When the GL receives M1 from v:
    Decrypts M1 using the symmetric key K and retrieves
    the message M;
    Checks the timestamp ts;
    Verifies the signature using public key PUv of v;
    Aggregates (M) with other messages sent by other vehicles;
    Computes M2 = (GL, RSU, E((M, ts), K), SIGGL);
    Sends M2 to RSU;
    // K is the symmetric key established between the GL and
    // the RSU when it entered the RSU's region.
When the RSU receives M2 from GL:
    Decrypts M2 using the symmetric key K and retrieves
    the message M;
    Checks the timestamp ts;
    Verifies the signature using public key PUGL of GL;
    Aggregates (M) with other messages sent by other GLs;
    Disseminates the message to the appropriate regions through
    other RSUs as well as vehicles in its region through the GLs.
```

Certificate Revocation List (CRL) distribution and certificate revocation process. Misbehaving vehicles can send malicious messages to other vehicles; these misbehaving vehicles should be detected and punished. IEEE 1609.2, the standard for Wireless Access in Vehicular Environments (WAVE)—Security Services for Applications and Management Messages [22], has specified that the vehicle must be authenticated using certificates issued by the TA and defined the CRL that contains the list of the revoked certificates that are updated timely and disseminated in the vehicular network. Once the CRL is distributed to the vehicles, it can compare the certificate of a vehicle with the list and determine if it has been revoked.

In our scheme, the DMV will manage and maintain the updated CRL. The DMV will distribute the CRL to the RSUs, which, in turn, will distribute them to all vehicles in their region directly or through the GLs, if the GLs have been selected. The RSUs and GLs always check the authenticity of the vehicles using the CRL. If a vehicle is found to be malicious, the RSU sends the certificate information of the vehicle to the DMV. Then, the DMV adds the certificate to the CRL and distributes the updated CRL to all RSUs. Note that vehicles only communicate either with the RSU or the GL and that no communication between themselves

occurs, which reduces the communication and computation overhead. We do not address the problem of detecting malicious vehicles. Many researchers have addressed the malicious vehicle detection problem in VANETs [23,24]. Any of those schemes can be used to detect malicious vehicles.

3.3. Some Optimizations for Our Approach

In our scheme, when a vehicle v enters the region of an RSU, it obtains a symmetric key K through the $Accept$ message $M_2 = (\text{“}Accept\text{”}, K, ts)$ from the RSU for establishing secure communication between v and the RSU (please see Algorithm 1). This key K is used by v to encrypt messages and send them to the RSU in the absence of GLs; this key is also used by the RSU to send messages, as well as $CRLs$, securely to v, in the absence of GLs. To reduce this overhead caused by sending unicast messages, the RSU can attach a group key GK to the accept message as $M_2 = (\text{“}Accept\text{”}, GK, K, ts)$; then, GK can be used by the RSU to broadcast (instead of unicasting) securely the $CRLs$ as well as other messages to all vehicles in its region. Similar optimizations can be performed in Algorithm 3 when a GL assigns a symmetric key K to a vehicle v through the message $M_2 = (\text{“}Accept\text{”}, K, ts)$.

4. Results

In our scheme, the encryption and the signature are fundamental security mechanisms used to resist impersonation, eavesdropping, replay, and modification attacks. The message that is sent by a vehicle v to its GL to be modified must be decrypted, modified, and then encrypted by an attacker using the $v's$ shared symmetric key. To decrypt the message, the attacker needs the symmetric key shared between the v and GL, which is not available to the attacker, thus making it impossible to modify the message. Replay attacks are prevented using timestamps. In our scheme, an attacker cannot generate a valid signature of other vehicles because the attacker does not know the private key of the vehicle. As a result, an attacker cannot send a malicious signed message without being detected.

Our scheme is secure against impersonation attacks: To perform an impersonation attack, the attacker should be able to obtain the private key PR_v of a legitimate vehicle v, which the attacker does not possess. In addition, an attacker cannot impersonate a vehicle v, as the message encrypted using a shared symmetric key K between v and GL (or between v and the RSU) cannot be decrypted without using K, which the attacker does not possess.

Our scheme preserves privacy—an attacker cannot discover the vehicle's identity: Vehicles are assigned pseudo-IDs. A vehicle never uses its real ID in any communication. This prevents discovering the real identity of the vehicle and prevents attackers from linking messages from the same vehicle using multiple pseudonyms. During registration, a vehicle is assigned a set of pseudonyms and associated certificates. Vehicles can use any of the pseudonym-changing strategies presented in the literature [21,25] to change pseudonyms. Therefore, the privacy of vehicles is preserved.

Communication and Computation Overhead: In our scheme, if the density of vehicles present in an RSU's region is low, it does not select GLs. If the density of vehicles in its region is high, then the RSU selects GLs from the vehicles to help the RSU with authenticating messages. The GLs are responsible for authenticating, aggregating, and forwarding messages received from vehicles from its sub-region. Thus, an RSU only needs to authenticate and process messages that come from the GLs. Therefore, the communication and computation overhead for an RSU is reduced. Note that an RSU sends messages to vehicles in its region through GLs; vehicles only need to authenticate messages received from its GL if the density of vehicles is high, and not from other vehicles, so the communication and computation overhead is low for the vehicles as well.

Figure 2 shows a comparison of the total communication cost of our scheme and that of the SEMA scheme [26], in terms of the number of messages exchanged between an RSU and the vehicles in its region. For the purpose of comparison, vehicle density within the region of an RSU is assumed to be high when the number of vehicles in its region is 1000 or more, and the average number of messages exchanged between a vehicle v and RSU

is 2; otherwise, we assume that the density is low. Figure 2 shows the average number of messages exchanged between an *RSU* and vehicles in its region with this assumption; if the number of vehicles is less than 1000 in its region, the *RSU* authenticates and processes messages received from all vehicles within its region; if there are more than 1000 vehicles in its region, the *RSU* needs to authenticate messages that comes from the *GL*s only. As a result, in our scheme, the communication cost is lower on the *RSU* side. For example, if there are only 400 vehicles present in the region of an *RSU*, the *RSU* will authenticate the same number of messages (which is $400 * 2 = 800$ messages) in our scheme and in the SEMA scheme [26]. For comparison purposes, to compute the number of *GL*s needed in an *RSU*'s region, we assume that a predefined threshold is 100 for each *GL*; i.e., if there are 1000 vehicles, the number of *GL*s needed is ($\lceil (1000/100) \rceil = 10$) and the number of messages exchanged between the *GL*s and the *RSU* would be ($\lceil (1000/100) \rceil * 2 = 20$) under our scheme, whereas under SEMA [26], the number of messages exchanged would be ($1000 * 2 = 2000$). Therefore, the total communication cost increases significantly with the increase of the number of vehicles under SEMA [26]. On the contrary, under our scheme, the communication cost is significantly lower. This is primarily because message collection overhead is shared by selected vehicles (GLs) in the RSU's region.

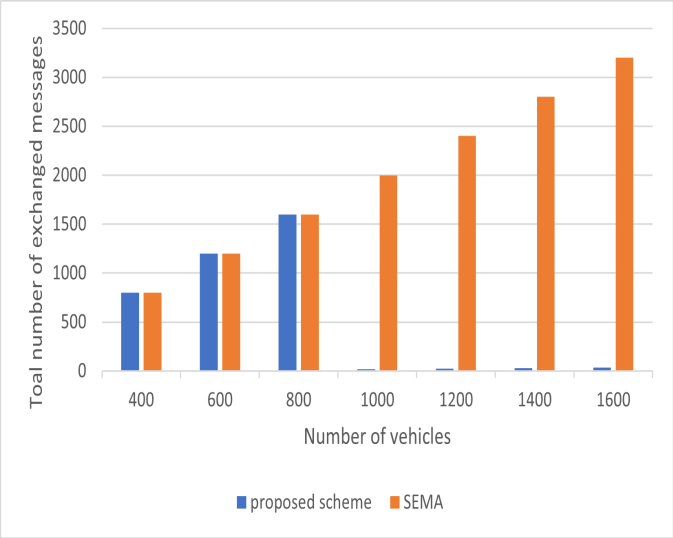

Figure 2. Communication overhead comparison.

We analyzed the computation overhead associated with encryption and authentication using a Toshiba computer with an Intel i3 quad-core processor with 2.50-GHZ clock frequency and 6 gigabytes of memory, running Windows 8.1 operating system. The public key cryptography-based signature and encryption scheme are based on RSA (Rivest–Shamir–Adleman) cryptography. Following are some notations used for presenting our results: time for computing RSA-based signatures (T_{sign}); time for signature verification (T_{verify}); time for encrypting a message using a public key (T_{EPU}); time for decrypting the message using a private key (T_{DPR}); time for encrypting a message using a symmetric key (T_{EK}); time for decrypting a message using a symmetric key (T_{DK}). We used the AES (Advanced Encryption Standard) to encrypt and decrypt the messages using a symmetric key. The execution time of the above operations is presented in Table 2. We used a message size of 39 bytes, as specified in the IEEE 1609.2 standard, for the encryption and the corresponding decryption operations.

Computation Overhead on *GL*: The *GL* is responsible for collecting, authenticating, and aggregating messages received from vehicles in its sub-region and forwarding them to the *RSU*. Figure 3 shows the computation overhead incurred by a *GL* for decrypting and verifying the signature of messages received from the vehicles in its sub-region as well encrypting and signing those messages for sending them to the *RSU* for a number of messages ranging from 50 to 500.

Computation Overhead for *RSU*: Figure 4 shows a comparison of the computation overhead between our scheme and SEMA [26] at an *RSU* for a varying number of signature verifications. Our scheme incurs significantly lower overhead compared to SEMA [26]. This is due to the use of the *GL*s, which help the *RSU* with the authentication and aggregation process of the messages sent by vehicles. For example, when the number of signatures reaches 1400, the overall cost is approximately 7 ms for the scheme in [26], whereas it is only 0.7 ms for our scheme.

Table 2. Execution time for different operations (milliseconds).

Operation	Time
T_{sign}	0.06
T_{verify}	0.005
T_{EPU}	1.274
T_{DPR}	2.654
T_{EK}	1.166
T_{DK}	2.128

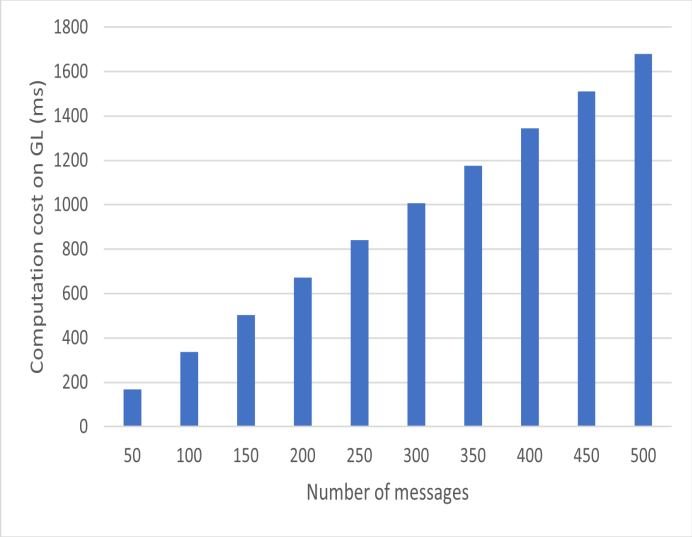

Figure 3. Total computation time at a *GL* for various numbers of messages.

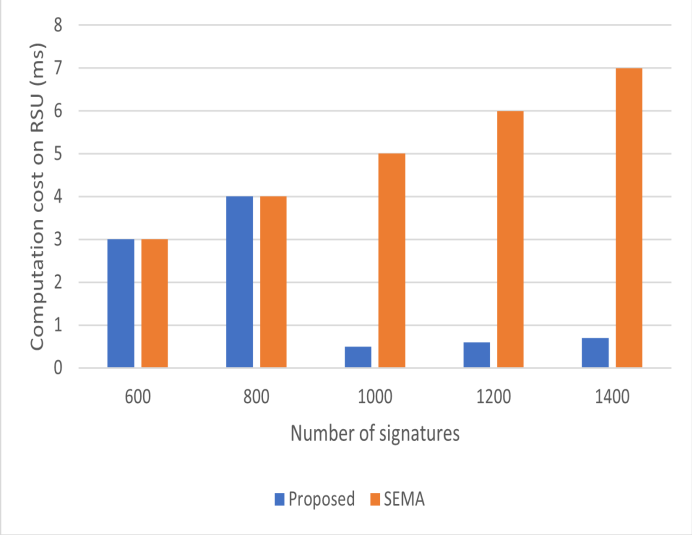

Figure 4. Computation time at the RSU.

5. Conclusions and Discussion

In this paper, we presented a low-overhead *RSU*-aided message authentication and dissemination scheme. In this scheme, when the overhead for collecting, authenticating, aggregating, and disseminating messages increases for an *RSU*, the *RSU* can designate some of the vehicles in its region as Group Leaders and make them share the overhead involved in authenticating, aggregating, and disseminating messages. Thus, this scheme helps the *RSU*s with reducing the computation and communication overhead related to collecting, authenticating, aggregating, and disseminating messages. We have also shown that our scheme is privacy-preserving and secure and resilient to various attacks. We also analyzed and compared the communication and computation overheads of our scheme with an *RSU*-aided approach for authentication and message dissemination.

Author Contributions: Both authors have contributed equally to all parts of the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Al-Sultan, S.; Al-Doori, M.; Al-Bayatti, A.; Zedan, H. A comprehensive survey on vehicular Ad Hoc network. *J. Netw. Comput. Appl.* **2014**, *37*, 380–392. [CrossRef]
2. Mistareehi, H.; Manivannan, D. Classification, challenges and critical comparison of proposed solutions for vehicular clouds. *Int. J.-Next-Gener. Comput.* **2019**, *10*, 1–18.
3. Rawashdeh, Z.; Mahmud, S. Intersection collision avoidance system architecture. In Proceedings of the 5th IEEE Consumer Communications and Networking Conference, Las Vegas, NV, USA, 10–12 January 2008.
4. Aung, N.; Zhang, W.; Dhelim, S.; Ai, Y. T-Coin: Dynamic traffic congestion pricing system for the internet of vehicles in smart cities. *Information* **2020**, *9*, 149. [CrossRef]
5. Aung, N.; Zhang, W.; Dhelim, S.; Ai, Y. Accident prediction system based on hidden markov model for vehicular ad-hoc network in urban environments. *Information* **2018**, *9*, 311. [CrossRef]
6. Abuashour, A.; Kadoch, M. Control overhead reduction in cluster-based VANET routing protocol. In *Ad Hoc Networks. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*; Springer: Cham, Switzerland, 2018; Volume 223, pp. 106–115.
7. Yassein, M.; Mistareehi, H. Improvement on the lifetime of the WSN using energy efficiency saving of leach protocol (New Improved LEACH). *Sensors Transducers J.* **2011**, *130*, 142–153.
8. Zhang, X.; Li, Y.; Miao, Q. A cluster-based broadcast scheduling scheme for mmWave vehicular communication. *IEEE Commun. Lett.* **2019**, *23*, 1202–1206. [CrossRef]
9. Singh, R.; Saluja, D.; Kumar, S. Graphical approach for V2V connectivity enhancement in clustering-based VANET. *IEEE Wirel. Commun. Lett.* **2021**, *10*, 6. [CrossRef]
10. Jo, H.; Kim, I.; Lee, D. Reliable cooperative authentication for vehicular networks. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 1065–1079. [CrossRef]
11. Lin, X.; Li, X. Achieving efficient cooperative message authentication in vehicular ad hoc networks. *IEEE Trans. Veh. Technol.* **2013**, *62*, 3339–3348.
12. Cheng, H.; Liu, Y. An improved RSU-based authentication scheme for VANET. *J. Internet Technol.* **2020**, *21*, 1137–1150.
13. Zang, C.; Lin, X.; Lu, R.; Ho, P. RAISE: An efficient RSU-aided message authentication scheme in vehicular communication networks. In Proceedings of the IEEE International Conference on Communications (ICC), Beijing, China, 19–23 May 2008.
14. Arkian, H.; Atani, R.; Diyanat, A.; Pourkhalili, A. A cluster-based vehicular cloud architecture with learning-based resource management. *J. Supercomput.* **2015**, *71*, 1401–1426. [CrossRef]
15. Dua, A.; Kumar, N.; Das, A.; Susilo, W. Secure message communication protocol among vehicles in smart city. *IEEE Trans. Veh. Technol.* **2018**, *67*, 4359–4373. [CrossRef]
16. Chaqfeh, M.; Mohamed, N.; Jawhar, I.; Wu, J. Vehicular cloud data collection for intelligent transportation systems. In Proceedings of the IEEE Smart Cloud Networks and Systems, Dubai, United Arab Emirates, 19–21 December 2016.
17. Azees, M.; Vijayakumar, P.; Deboarh, L. EAAP: Efficient anonymous authentication with conditional privacy-preserving scheme for vehicular adhoc networks. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2467–2476. [CrossRef]
18. Liu, Y.; Wang, Y.; Chang, G. Efficient privacy-preserving dual authentication and key agreement scheme for secure V2V communications in an IoV paradigm. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2740–2749. [CrossRef]
19. Yang, Y.; Zhang, L.; Zhao, Y.; Choo, K.; Zhang, Y. Privacy-preserving aggregation-authentication scheme for safety warning system in Fog-Cloud based VANET. *IEEE Trans. Inf. Forensics Secur.* **2022**, *17*, 317–331. [CrossRef]

20. Zhang, L.; Wu, Q.; Solanas, A.; Domingo-Ferrer, J. A scalable robust authentication protocol for secure vehicular communications. *IEEE Trans. Veh. Technol.* **2010**, *59*, 1606–1617. [CrossRef]
21. Mistareehi, H.; Islam, T.; Manivannan, D. A secure and distributed architecture for vehicular cloud. *Internet Things* **2021**. [CrossRef]
22. *1609.2-2016*; IEEE Standard for Wireless Access in Vehicular Environments–Security Services for Applications and Management Messages; IEEE: Piscataway, NJ, USA, 2016; pp. 1–240.
23. Nguyen, V.; Lin, P.; Hwang, R. Enhancing misbehavior detection in 5 g vehicle-to-vehicle communications. *IEEE Trans. Veh. Technol.* **2020**, *69*, 9417–9430. [CrossRef]
24. Gyawali, S.; Qian, Y.; Hu, R. A privacy-preserving misbehavior detection system in vehicular communication networks. *IEEE Trans. Veh. Technol.* **2021**, *70*, 6147–6158. [CrossRef]
25. Ying, B.; Makrakis, D. Pseudonym changes scheme based on candidate-location-list in vehicular networks. In Proceedings of the IEEE International Conference on Communications (ICC), London, UK, 8–12 June 2015.
26. Wang, P.; Liu, Y. SEMA: secure and efficient message authentication protocol for VANETs. *IEEE Syst. J.* **2021**, *15*, 846–855 . [CrossRef]