

Secure and Dynamic Route Navigation Through RSU-Based Authentication in IoV for Smart City

Bimal Kumar Meher[✉], *Member, IEEE*, Ruhul Amin[✉], *Senior Member, IEEE*, Mohammad Abdussami[✉],
Muhammad Khurram Khan[✉], *Senior Member, IEEE*, Md Abdul Saifulla[✉], and Sanjeev Kumar Dwivedi[✉]

Abstract—One of the significant services provided by IoV in Smart cities is vehicular navigation. Drivers often find it difficult and time-consuming to complete their trip in a crowded city without real-time knowledge about the traffic and road conditions. So, a proper routing mechanism can help drivers reach their destination in minimum time and with less fuel consumption. However, it has been found that such protocols often face security challenges. In this paper, we have proposed an authenticated navigation scheme with the help of pseudonym-based asymmetric-key cryptography that discovers and secures the route to the destination in real time. The architecture embodies a geolocation provider (GLP) to get the possible static routes to a particular destination. Further, it uses the message-forwarding capability of RSUs to develop a dynamic route, after receiving feedback from the respective RSUs about the traffic conditions. While doing so, this protocol ensures proper message integrity, anonymity, unlinkability and robust protection from important security threats. Our approach ensures minimal end-to-end delay and efficient real-time route finding from a source to a destination with no extra overhead on the vehicles. We have simulated our authentication protocol using the Scyther simulator and found it safe from various adversarial attacks.

Index Terms—IoV security, vehicular ad-hoc network, authentication, navigation, ElGamal, Scyther.

I. INTRODUCTION

EFFICIENCY in smart city vehicular transportation has been improved significantly due to the merger of the

Internet of Things (IoT) and the vehicular ad-hoc networks (VANET). This amalgamation of technologies has elevated vehicular communication to the next level, known as the Internet of Vehicles (IoV). It has the potential to revolutionize future mobility since it has successfully overcome the limitations of VANET such as low network coverage area, limited mobility, and restriction on the number of vehicles [1]. Besides supporting safe transportation, traffic management, and real-time data dissemination, it has the capability to deliver better driver/user comfort, accident reduction, pollution control, and fuel saving. The incorporation of various social and user-centric services for commuters is going to open vast commercial opportunities. For example, to make the IoV more productive and user-centric, the idea of a vehicular social network (VSN) has evolved recently [2], [3], [4]. A VSN incorporates the socialization aspect, where users can exploit their mobility and generate customized location-dependent information. For example, a user can prepare a travel guide of his location and share it with other users for tourism promotion. Therefore, IoV is all set for a robust future cooperative communication system among the vehicles, road-side units (RSU), cloud servers, and smart gadgets like sensors, cameras etc. (installed on the road or in the vehicle). The connectivity on the road is primarily due to vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) connectivity. Data acquired by vehicles are transmitted either to an RSU or a server for processing. The processed output is made available to vehicles as driving recommendations. It helps a driver on the road to be more aware of his surroundings and enables him to adapt dynamic decisions on the go. Therefore, IoV has emerged as a potential candidate for an intelligent transportation system (ITS) for ensuring road safety, congestion control, ride-time-saving, and air-pollution control, besides facilitating socialization on the go and location-based services like parking/petrol pump/restaurant information, tourist information etc. [5].

Fig. 1 shows a typical IoV model with entities like cloud server, RSUs, vehicles (with OBU) and pedestrians/cyclists with smartphones. Besides vehicle-to-vehicle communication, pedestrians/cyclists with smartphones can also share valuable information with the vehicles. Here, the RSUs directly communicate with the cloud server after receiving the information from the vehicles. Since the cloud server is a centralized facility, it often encounters serious problems elaborated next. Firstly, it faces the issue of high computational latency due to the loads exerted by RSUs. Since all the RSUs are connected directly to the server, it often fails to provide real-time services to the

Received 8 August 2023; revised 12 April 2025; accepted 12 April 2025. Date of publication 22 April 2025; date of current version 25 August 2025. The work of Muhammad Khurram Khan was supported by King Saud University, Riyadh, Saudi Arabia under Project RSP2025R12. This work was supported by the International Institute of Information Technology, Naya Raipur. Recommended for acceptance by Dr. Dusit Niyato. (Corresponding authors: Ruhul Amin; Muhammad Khurram Khan.)

Bimal Kumar Meher is with the Department of Computer Science and Engineering, Silicon University, Odisha, Bhubaneswar 751024, India (e-mail: bimalmeher@gmail.com).

Ruhul Amin is with the Computer Science and Engineering Department, International Institute of Information Technology, Naya Raipur 493661, India (e-mail: amin_ruhul@live.com).

Mohammad Abdussami is with the Department of Computer Science and Engineering, SRM University AP, Amaravati, Andhra Pradesh 522240, India (e-mail: javid.sami@gmail.com).

Muhammad Khurram Khan is with the King Saud University, Riyadh 11653, Saudi Arabia (e-mail: mkhurram@ksu.edu.sa).

Md Abdul Saifulla is with the School of Computer and Information Sciences, University of Hyderabad, Hyderabad 500046, India (e-mail: saifullah@uohyd.ac.in).

Sanjeev Kumar Dwivedi is with the Centre for Artificial Intelligence, Madhav Institute of Technology and Science, Deemed University (MITS-DU), Gwalior 474005, India (e-mail: sanjeevswivedi131988@gmail.com).

Digital Object Identifier 10.1109/TNSE.2025.3563297

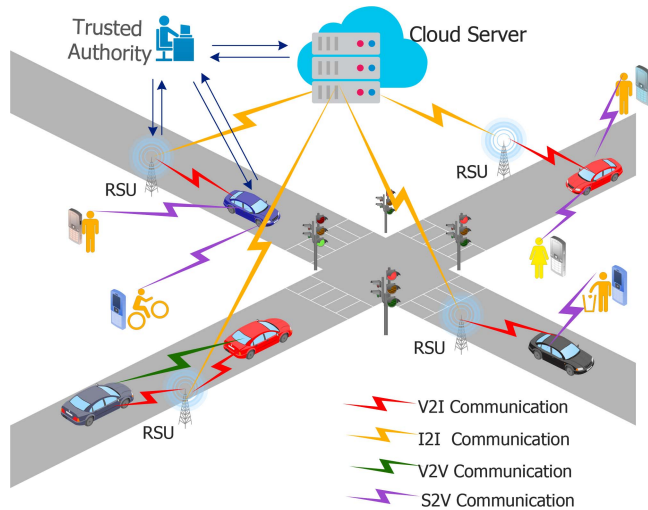


Fig. 1. A typical IoV system model.

drivers and passengers. It may lead to gross failure in fulfilling the objective of the IoV. Secondly, breakdown of the cloud server leads to complete system failure unless there is an alternate arrangement. Thirdly, security vulnerability would play havoc in the centralized server, thus causing extensive damage to human lives. To address the first and the second issues, fog computing can be used to minimize the load on a single centralized cloud server [6], [7].

To address the third issue, it is crucial to separate server data security from message authentication. In IoV, where messages [8] are dynamically exchanged among vehicles, RSUs, and service providers, strong authentication is vital. A major challenge is protecting users' long-term identity and location, [9], [10] as their exposure can lead to severe attacks and safety risks. To preserve privacy, researchers have proposed pseudonym-based authentication, where short-term tokens replace real identities. These pseudonyms can be dynamically generated or pre-stored. While they protect privacy, mechanisms must exist to trace and revoke identities of malicious users and reward trustworthy ones. This work proposes a comprehensive, pseudonym-based route navigation scheme to ensure secure and authenticated vehicular communication.

A. Our Contributions

The main contributions of this paper are as follows:

- We propose a secure and dynamic route navigation framework through RSU-based authentication for the Internet of Vehicles.
- Our design uses a geo-location provider (GLP) and a set of RSUs to realize the scheme without involving the vehicles directly. Therefore, it eliminates the dependency of RSUs on the vehicles for collecting traffic-related information and reduces delay in authentication.
- Navigation queries and replies between the RSUs are duly protected by using ElGamal-based encryption, digital signature and public key certificates.

- The computation and communication costs are also reduced in comparison with other relevant schemes since our scheme doesn't use computation-intensive ECC point multiplication or Bilinear pairing operations.
- Our scheme addresses security issues like driver anonymity, session key secrecy, unlinkability, and adversarial attacks like collusion, Sybil, MITM, replay, and masquerading efficiently in comparison to other schemes.
- The GLP in our framework acts as a trusted security agent between the navigation route requester (vehicle/RSU) and the route finders (RSUs) to support the above claim.
- Security analysis and verification by using the Scyther tool proves the robustness of our design.

B. Organization of the Paper

The rest of our paper is organized as follows: Some of the prominent research works related to our problem are highlighted in Section II. In Section III, we present our proposed protocol with a brief overview of the underlying mathematical concepts used in our protocol. Detailed explanation and working is illustrated in Section IV. Informal security analysis along with the simulation result by using the Scyther tool is presented in Section V. Performance comparison of our scheme with some recent works is tabulated in Section VI followed by the conclusion in Section VII.

II. RELATED WORK AND SECURITY GOALS AND ATTACKS

A. Related Work

Chim et al. [11] has presented a navigation scheme that guides the drivers to reach the destination in a vehicular ad-hoc network (VANET). It uses real-time information to compute a better route and also uses suitable source authentication. Anonymous credential is used to protect the privacy of the drivers. Their scheme uses bilinear maps and proxy re-encryption to provide the necessary privacy and authentication to the messages and communicating parties.

Ni et al. [12] has proposed a real-time navigation system using vehicular crowdsourcing. The RSUs cooperatively find an optimal path to the destination from the crowdsourced data by the vehicles in their coverage areas. So, a vehicle retrieves the navigation information from the RSUs on entering their coverage area, until it reaches the destination. They have used message-lock encryption and key-homomorphic signatures for the safety of the driver's location and identity. In addition, a trusted authority can trace the drivers' identities if they upload false traffic information.

Wang et al. [13] has proposed a fog-based VANET for real-time navigation. Fog nodes, in their paper, perform both the crowdsourcing job and route finding in a dynamic traffic condition. The privacy and authentication are achieved with the help of ElGamal, AES, and group signature schemes.

Li et al. [14] has proposed a privacy-preserving navigation scheme, which supports similar queries in navigation services. They transform the typical navigation approach into a traffic congestion querying approach. That means drivers query the

traffic congestion along the navigation route. They realize their work by using privacy-preserving multi-keyword fuzzy search and constructing weighted proximity graphs. They claim that their scheme protects location and route privacy, and defends against spurious reporting, and collusion attacks.

Ni et al. [15] has proposed a smart parking navigation system using Bloom filters. It enables a cloud to guide vehicles to vacant parking spaces based on real-time parking information. The drivers can query for parking spots to the cloud, and retrieve the encrypted navigation results from the RSUs.

Zhou et al. [16] has devised a lightweight real-time traffic navigation scheme for cloud-assisted VANETs. It facilitates the prediction of an optimal driving route, in terms of the shortest time from source to destination. It also evaluates the auto-regression moving average (ARMA) model with spatiotemporal correlations of high accuracy and efficiency. Therefore, they claim that, neither users' private location information nor the navigation result will be disclosed, even in the presence of a colluding semi-trusted cloud server.

Baruah et al. [17] has proposed an intelligent vehicle navigation system by using a distributed approach. It retrieves the navigation information without revealing the destinations to the RSU or the TA. They use the bilinear pairing concept to realise their scheme.

B. Security Goals and Attacks

- 1) *Message integrity*: The navigation query or response messages must be tamper-proof to ensure message integrity. Any unauthorized change to the content of these messages could compromise the system's security. Therefore, the receiver should check the integrity of the query message before accepting it and sending a response. Techniques such as hash functions and digital signatures are employed to maintain the integrity of the message.
- 2) *Anonymity*: This feature prevents the real identities of the senders and receivers from being revealed to each other. It is essential to keep the identity of the participating entities safe and undisclosed in an insecure environment like IoV. Therefore, a pseudonymous identity assigned by TA (often called a pseudonym) is used to achieve anonymity in vehicular communication.
- 3) *Unlinkability*: An attacker may be able to associate a message with its sender or receiver, potentially exposing the real identities of drivers and passengers, as well as their travel destinations. It poses a significant security threat to both parties. Therefore, incorporating the unlinkability feature can prevent an attacker from discovering the previous communications between the senders and receivers, thereby ensuring the security of the IoV system. It is typically achieved through the use of dynamic pseudonyms that change over time.
- 4) *Replay Attack*: An attacker intercepting a navigation message between two communicating entities and replaying it can lead to significant confusion in a vehicular network. To prevent such attacks, the sender must implement proper time-stamping and ensure the use of a unique session ID.

The receiver should verify these elements to discard any duplicate messages.

- 5) *Impersonation Attack*: An attacker may impersonate an authentic entity and send a response to the navigation query by a vehicle. If such malicious activity goes unnoticed, it could mislead the driver onto the wrong route. This may lead to severe traffic jams, inconvenience for commuters, and excessive fuel consumption, among other consequences.
- 6) *Sybil Attack*: Attackers may try to infiltrate the vehicular network by creating fake identities. If successful, they can gain control of the network and disrupt the entire system. Therefore, identity validation is an effective method for preventing such attacks. A trusted authority (TA) plays a crucial role in neutralizing these types of attacks.

III. OUR PROPOSED PROTOCOL

A. Mathematical Background

Our proposed protocol utilizes cryptographic operations, including encryption, decryption, and digital signature from the ElGamal cryptosystem, along with cryptographic hash function. Common operations such as multiplication and exponentiation are based on a multiplicative group G with a generator g . The security of this type of cryptosystem is based on the computationally hard mathematical problem called discrete logarithms problem (DLP) and is defined for multiplicative groups as follows:

Let G be a group and g is a generator of G , then every element e in G can be written as g^x for some x . So, the discrete logarithm to the base g of e in G is x . The discrete logarithm problem (DLP) is defined as follows:

Given a group G , a generator g and an element e , find x such that, $g^x = e$. It is to be noted that finding such x is not always hard, since it depends on the property of the underlying groups. Therefore, discrete logarithm-based cryptosystems use the group Z_p^* , where p is a large prime number of the order of 1024 bits to keep it safe from attacks.

ElGamal cryptosystem is a public key cryptosystem based on the computational hardness of DLP and provides efficient algorithms for key generation, encryption, and decryption. ElGamal digital signature algorithm is another strong candidate for signing a message to achieve authenticity and integrity. It has also three steps involving key generation, signature generation, and signature verification.

A schematic diagram of our navigation model is depicted in Fig. 2. It contains the entities like TA, RSUs, vehicles, and GLP. We now briefly explain the functionality of these entities.

B. Entity Definition

- 1) *TA*: The trusted authority (TA) is a secured entity that preserves all private information in its database by applying a secret key hashing technique. It ensures the privacy of user information in case of a data leak from the TA's server. Information stored with the TA is usable for tracking the vehicles' real identity from the pseudonym

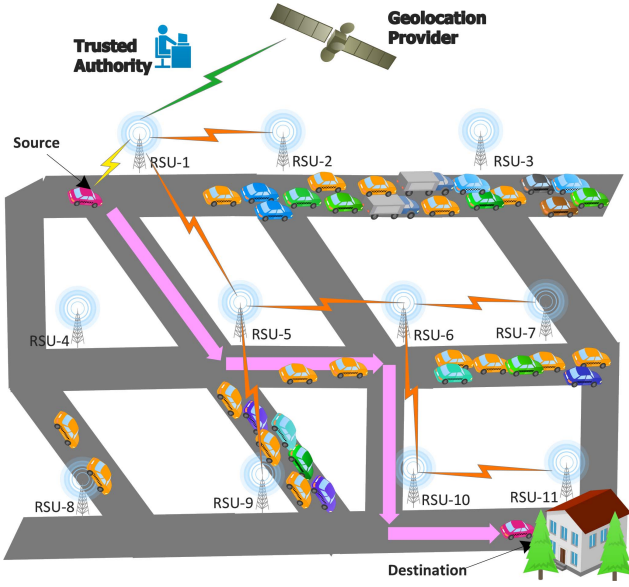


Fig. 2. Route selection in our navigation model.

identity in case of disputes. In addition to this, the TA also initializes system parameters and issues secret credentials and digital certificates used by other entities. Moreover, using multiple certificates in place of one ensures privacy against tracking.

- 2) *GLP*: Geo-location provider (GLP) is a satellite-based system that provides possible static routes to users according to their geographical location. It has the RSU-based location map of the region under operation. Therefore, it knows a specific geo-location and can find the feasible routes from a source to a destination using RSUs as the hop information.
- 3) *RSU*: Road side units (RSU) are smart electronic devices deployed along the road side and act as key facilitators of IoV. They help in traffic control, road safety, and emergency response, and also provide additional services such as entertainment, social activity, information about petrol pumps, restaurants, etc. Therefore, RSUs have embedded processors, V2X modules, cameras, radar, and other auxiliary units. They also connect to the cloud for efficient data storage and distribution over the vehicular network. Before starting work, they are first registered with the TA and receive valid credentials. Then, they start sending and receiving messages to/from other entities like GLP, vehicles, and other RSUs.
- 4) *Vehicle*: Each vehicle receives a unique identity and secret credentials after successful registration with the TA. The OBU (On-Board Unit) fitted in the vehicle permits the transmission of messages between the vehicles (V2V) and between the vehicle and RSU (V2I). The messages can be either traffic-related or other messages related to user comfort/entertainment. Therefore, each OBU has a tamper-proof memory to store confidential information such as secret credentials, location and time information

of an event like an accident, information about a petrol pump, restaurant etc.

We use the notations provided in Table I to explain the working of the proposed protocol.

C. Working

In our protocol, we use pseudonyms to hide the actual identity of the vehicle or its user, thus achieving conditional anonymity [18], [19], [20]. Pseudonyms are either provided by the TA (preloaded with the vehicle) or self-generated by the vehicles dynamically. Since pseudonym has an expiry period, an attacker cannot use them afterwards, thus preventing replay attacks. Moreover, the frequent change of pseudonyms in a pre-defined order helps to achieve untraceability and unlinkability.

1) *System Initialization*: The TA is responsible for initializing the system with some important parameters. Subsequently, these parameters are made available to RSUs and vehicles for authentication and communication. The TA chooses a group G of order q with generator g . Then it chooses a random number $s \in \mathbb{Z}_q^*$ as the master secret and computes its public key $PU_{TA} = g^s$. It also chooses one-way hash function $h(\cdot)$ to be used in subsequent operations. It also creates its own digital certificate $Cert_{TA} = \langle TA, PU_{TA}, SIG_{PR_{TA}}(TA, PU_{TA}) \rangle$ and publishes the system parameters as $\langle G, q, g, h(\cdot), Cert_{TA} \rangle$.

2) *Geolocation Provider Registration*: The GLP sends its identity proof to the TA for registration through a secure channel. It may use any secure communication protocol like TLS as it has a domain name received from the competent authority. It is essential because TA has to keep a record of all the entities (GLP, RSU, vehicle) taking part in a geographical area to ensure the security and privacy of the IoV. The GLP server receives a unique ID from the TA as $PID_{GLP} = h(s || ID_{GLP})$ and stores it in its memory safely for future use. In addition, it receives the private key $PR_{GLP} = g^l$, where l is a random number selected by the TA and the public key $PU_{GLP} = PR_{GLP}^s$.

3) *RSU Registration*: The RSU sends its identity (ID_{RSU_j}) proof to the TA for registration through a secure channel along with the location code (LC_j). TA creates a pseudo-identity $PRID_j = h(s || ID_{RSU_j} || LC_j)$. It also chooses a private key PR_{RSU_j} and generates its public key $PU_{RSU_j} = g^{PR_{RSU_j}}$. Then creates a certificate $Cert_{RSU_j} = \langle PRID_j, LC_j, PU_{RSU_j}, SIG_{PR_{TA}}(PU_{RSU_j}, PRID_j, LC_j) \rangle$. Finally, it sends $\langle Cert_{RSU_j}, Cert_{GLP}, PR_{RSU_j} \rangle$ to RSU_j through a secure channel. After receiving it, RSU_j keeps the information in a tamper-proof device and publishes its public key.

4) *Vehicle Registration*: Each vehicle sends its identification (ID) proof to the TA through a secure channel to be registered in the IoV system. It is a mandatory one-time verification of the essential documents and identities of the entities prior to registration. This ensures the optimal correctness of the information entered into the database for future reference. After successful verification, TA provides a tamper-proof device (TPD) with its master key stored inside. Usually, TPD is also known as a hardware security module (HSM) different from the OBU. TA also assigns a pseudo-identity $PID_{V_i} = h(s || ID_{V_i})$

TABLE I
NOTATIONS USED IN OUR PROPOSED PROTOCOL

Notations	Meaning
GLP, V_i	Geolocation Provider, Vehicle i
R_j, TA	Road Side Unit j , Trusted Authority
ID_{V_i}, ID_{R_j}	Identity of V_i , Identity of R_j
$PID_{V_i}, PRID_j$	Pseudo-identity of V_i , Pseudo-identity of R_j
SND	Session-ID created by RSU
SID	Session-ID created by GLP
s, l, v	Random numbers
PR_x, PU_x	Private key of entity x , Public key of entity x
g, G	g is a generator of group G
$h(.)$	One-way hash function $h : \{0, 1\}^* \rightarrow G$
$ENC_{PU_x}(M_k)$	Encryption of message M_k with public key PU_x of entity x
$SIG_{PR_x}(M_k)$	Signature on message M_k with private key PR_x of entity x
$Cert_x, T$	Public key certificate of entity x , Timestamps

to each vehicle V_i and a TPD activation password PW_{V_i} . It chooses a random number $v \in Z_p^*$. Then it produces the private key $PR_{V_i} = g^v$ and the public key $PU_{V_i} = PR_{V_i}^s$. Then it produces its public key certificate as $Cert_{V_i} = \langle PID_{V_i}, PU_{V_i}, SIG_{PR_{TA}}(T || PID_{V_i} || PU_{V_i}) \rangle$. Then sends $\langle PID_{V_i}, PW_{V_i}, PU_{V_i}, PR_{V_i}, Cert_{V_i} \rangle$ to the vehicle through a secure channel along with the TPD. When the TPD is installed in the designated vehicle, the user of the vehicle activates it with PID_{V_i} and PW_{V_i} .

5) *Navigation Scheme*: Drivers often find it difficult to search for the optimal route from a source to a destination due to the huge volume of vehicular traffic and its dynamic behavior in a large city. Researchers have published a few research papers on finding sub-optimal solutions using AI/ML techniques because of the complexity level of the problem. Since RSUs and vehicles are directly involved in sending navigation messages for route discovery, there should be robust authentication to avoid falling prey to wrong route discovery. Therefore, a secure navigation scheme helps a driver find the correct route and minimize travel time and fuel consumption without compromising his safety and privacy. Fig. 2 shows our secure navigation scheme with its higher-level functionalities explained below:

- 1) When an RSU receives a navigation query in the form of a pair (source, destination) from an on-road vehicle, it first authenticates the vehicle and then the query message. On successful verification, it sends the query to the GLP to find the static routes to the destination of the query.
- 2) The GLP then finds some possible routes for the source-destination pair by following the shortest path algorithm like Dijkstra's and maps the geolocations/places to the respective zone RSUs. However, it does not reveal these routes to the RSU from which it received the query for safety reasons. Only sends an acknowledgment to that RSU.

- 3) Then GLP sends the RSU-based navigation query to the starting (neighboring) RSUs (assuming that there are multiple routes for a source-destination pair) for onward transmission to discover the dynamic route, based on the traffic conditions in their respective zones. The RSUs use one-to-one communication rather than broadcasting (flooding) to save communication bandwidth.
- 4) Each RSU en route keeps a copy of the route in its cache for future reference and sends the navigation query message to neighboring one-hop RSUs.
- 5) A one-hop RSU, after receiving the query, checks the destination RSU in the message. Then it searches its database to know whether the traffic/road/weather condition under its zone (radio range) is favorable for the vehicle to pass through or not. If so, it finds all the one-hop neighbors towards the destination from the route sketch it received from the upstream RSU. Then it prepares a similar message it received earlier and forwards it to the next one-hop RSUs. But, if the road is not in favorable condition, it simply ignores the query message. It would not impact the sender as it would go for retransmission after session time-out.
- 6) The previous step is repeated until the query message of the vehicle finally arrives at the destination RSU. Then it prepares the return navigation (reply) message and sends it back to the upstream RSU from which it received the query message. This process recursively followed until the final reply message reached the GLP. In case of more than one reply message received by the GLP, it keeps only one message based on the lowest hop count (least number of intermediate RSUs) and discards the rest.
- 7) Then the GLP replies to the queried RSU and delivers the navigation route for onward transmission to the queried vehicle.
- 8) Finally, the queried RSU sends the route navigation information back to the vehicle. If the vehicle does not receive the reply message within a time interval of δT , then it assumes that it is outside the radio range of the queried RSU or the query/reply message has been lost/delayed during transmission. So, the vehicle may decide to run a new navigation query in the radio range of the same RSU or a new RSU (if the vehicle has entered its zone) after the expiration of the time interval δT .

IV. AUTHENTICATION AND NAVIGATION

In our protocol, entities are first authenticated with the help of their certificates and digital signatures. The authenticity of the navigation scheme depends on the secure transmission of messages between entities. Since navigation relies primarily on three entities, we have focused on security and privacy issues associated with their communication as follows:

- 1) Secure communication between vehicle and the first RSU
- 2) Secure communication between the first RSU and the GLP
- 3) Secure communication between RSU_i and RSU_j (from the second RSU to the destination RSU)

We now present the details of secure navigation with the help of a simple scenario of the smart city as shown in Fig. 2. We assume that there are a total of 11 RSUs (R_1 to R_{11}) on the way from the source to the destination, as shown in the figure. Let a vehicle V_i moving near R_1 need navigation information to reach its destination near R_{11} with minimal discomfort/delay due to traffic conditions. Now, in the following subsection, we explain the secure communication steps that involve all incumbent entities along the trip.

A. Vehicle – RSU_1 Communication

Step 1: First, vehicle V_i sends the message $M_1 = \langle PRID_1, Cert_{V_i}, T, NQ, SIG_{PR_{V_i}}(PRID_1 || NQ || T) \rangle$ to R_1 from which it has received the beacon signal. $Cert_{V_i}$ is the public key certificate of V_i received earlier from TA during registration. Initially, NQ is a null string to denote the navigation query, and T is the timestamp.

Step 2: Then R_1 uses the public key of TA to verify the signature in the certificate. If the TA signature is valid, then it checks the signature of V_i by using the public key (PU_{V_i}) from the certificate of V_i . On successful verification, it knows that there is a navigation query from V_i .

Step 3: So, the RSU generates a session-ID (SND) for the navigation query. Then it encrypts the SND and $Cert_{GLP}$ using PU_{V_i} and sends it to the vehicle V_i . It also sends the timestamp of the message creation. So, the complete message becomes $M_2 = \langle PRID_1, PID_{V_i}, Cert_{R_1}, T, SIG_{PR_{R_1}}(PRID_1, PID_{V_i}, T), ENC_{PU_{V_i}}(SND, Cert_{GLP}) \rangle$. After verifying the signature of R_1 and the freshness of the message, the vehicle decrypts the encrypted portion to know the session-ID and public key of the GLP.

Step 4: The vehicle then prepares the final message to the RSU with the query $NQ = \langle S, D \rangle$, encrypted with PU_{GLP} , where S and D denote the start and finish point of a trip, respectively and then signs it with its private key. So, the message $M_3 = \langle PID_{V_i}, PRID_1, T, ENC_{PU_{GLP}}(PID_{V_i}, PRID_1, SND, NQ) \rangle$ along with the signature $SIG_{PR_{V_i}}(PID_{V_i}, PRID_1, T)$.

B. RSU_1 – GLP Communication

Step 1: After receiving the message along with the signature, RSU verifies the signature, but cannot decrypt the encrypted portion as that is encrypted with the public key of GLP. On successful verification, it removes the encrypted portion and sends a message to GLP using its ID, certificate, timestamp along with the encrypted portion received from the vehicle V_i . So, the message $M_4 = \langle PRID_1, Cert_{R_1}, PID_{GLP}, T, ENC_{PU_{GLP}}(PID_{V_i}, PRID_1, SND, NQ) \rangle$ along with the signature $SIG_{PR_{R_1}}(PRID_1, PID_{GLP}, T)$ is sent to the GLP.

Step 2: After verifying the certificate, GLP finds $PRID_1$ and checks the freshness of the message with the timestamp T . Then it decrypts the encrypted portion with its private key, gets the navigation credential $\langle PID_{V_i}, PRID_1, SND, NQ \rangle$ and sends an acknowledgment to RSU R_1 .

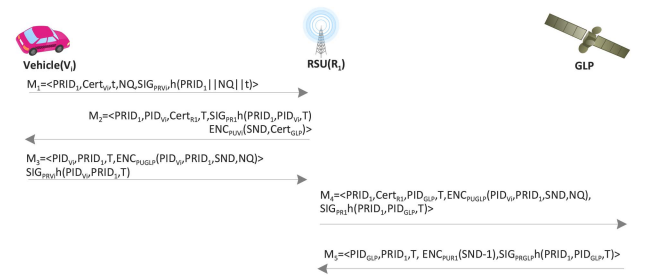


Fig. 3. Secure message communication involving Vehicle (V_i), RSU (R_1), and GLP.

A summary of the above procedure is shown in Fig. 3. Then the GLP finds all the possible static routes (RT_1, RT_2, \dots) for the starting point (S) and ending point (D) mentioned inside NQ . Then sends messages following the format $\langle RT_1, RT_2, \dots, SID(\text{new session ID}), T \rangle$ encrypted with the public key of the respective RSUs with its signature.

C. RSU_i – RSU_j Communication

Let us assume that GLP has discovered four routes $RT_1 = \langle R_1, R_2, R_6, R_7, R_{11} \rangle$, $RT_2 = \langle R_1, R_5, R_6, R_{10}, R_{11} \rangle$, $RT_3 = \langle R_1, R_5, R_9, R_{10}, R_{11} \rangle$, $RT_4 = \langle R_1, R_5, R_6, R_7, R_{11} \rangle$. Since, in all four routes, the next hop RSU is either R_2 or R_5 , it sends a message to R_2 and R_5 . It maps the old session-ID (SND) to a new session-ID (SID) and stores the mapping information in its database for future use when the final response of the navigation route returns. It also keeps the NQ , RSU-ID, and vehicle-ID in its database. So, in our example, it keeps the following credentials in its database: $\langle NQ, RSU-ID, Vehicle-ID, map(SND, SID), To = R_2, R_5 \rangle$. Then, GLP sends the following message and signature to RSU_2 and RSU_5 respectively. Message to R_2 : $\langle ENC_{PU_{R_2}}(NQ, RT_1, T, SID, Cert_{RSU_1}), SIG_{PR_{GLP}}(NQ, RT_1, T, SID) \rangle$. Message to R_5 : $\langle ENC_{PU_{R_5}}(NQ, RT_2, RT_3, RT_4, T, SID, Cert_{GLP}), SIG_{PR_{GLP}}(NQ, RT_2, RT_3, RT_4, T, SID) \rangle$. After receiving the message and signature, both R_2 and R_5 first decrypt the message using their respective private keys and then verify the signatures. On successful verification, both become aware of the navigation query NQ . Since R_2 faces traffic jams as shown in Fig. 2, it cannot help GLP to find a suitable route for the navigation query NQ . So, it remains silent without doing any further tasks. On the other hand, R_5 being in better condition can help GLP to find a route to R_{10} (destination as mentioned in NQ). So, it prepares a message and signature for its one-hop neighbors R_6 and R_9 as follows: Message to R_6 : $\langle ENC_{PU_{R_6}}(NQ, RT_2, RT_4, T, SID, Cert_{RSU_5}), SIG_{PR_{R_5}}(NQ, RT_2, R_4, T, SID) \rangle$. Message to R_9 : $\langle ENC_{PU_{R_9}}(NQ, RT_3, T, SID, Cert_{RSU_5}), SIG_{PR_{R_5}}(NQ, RT_3, T, SID) \rangle$.

It also keeps a copy of the information $\langle NQ, SID, R_6, R_9 \rangle$ in its database for future reference.

From the figure, it is clear that the road through R_9 faces a traffic jam problem and follows the same principle as that of R_2 . But R_6 is in better condition to help R_5 . So, it prepares a message and signature for its one-hop neighbors R_7 and R_{10} as follows:

It sends a message to R_7 : $\langle ENC_{PU_{R_7}}(NQ, RT_4, T, SID, Cert_{RSU_6}), \text{with signature } SIG_{PR_{R_6}}h(NQ, RT_4, T, SID) \rangle$, and also a message to R_{10} : $\langle ENC_{PU_{R_{10}}}(NQ, RT_2, T, SID, Cert_{RSU_6}), \text{with signature } SIG_{PR_{R_6}}h(NQ, RT_2, T, SID) \rangle$. It also keeps a copy in its database as follows: $\langle NQ, SID, R_7, R_{10} \rangle$

Since R_7 has a traffic jam condition, it remains silent. But, R_{10} , after receiving the message, checks it by verifying the TA signature in the certificate $Cert_{R_6}$ and then verifies the signature using the public key (PU_{R_6}) of R_6 . Then it prepares a message and signature for its one-hop neighbor R_{11} and sends a message $\langle ENC_{PU_{R_{11}}}(NQ, RT_2, T, SID, Cert_{R_{10}}), \text{with signature } SIG_{PR_{R_{10}}}h(NQ, RT_2, T, SID) \rangle$. It also keeps a copy in its database as follows: $\langle NQ, SID, R_{11} \rangle$

By checking NQ , R_{11} concludes that the query message has reached its destination and has to send the reply message back to its upstream RSU whose public key has just been retrieved from the message. So, it prepares a message as follows:

Message from R_{11} to R_{10} : $\langle ENC_{PU_{R_{10}}}(T, SID, R_{11}) \rangle$ and digest $\langle h(T||SID||R_{11}) \rangle$ to check the integrity of the message. Other RSUs just discard the message and the digest as they cannot decrypt it. After receiving the message, R_{10} knows that it is a reply message for some query it sent earlier. To know more about the query, it decrypts it with its private key and checks its database to see if it matches the navigation credentials: $\langle NQ, SID \rangle$ along with the partial navigation route $\langle R_{11} \rangle$ to whom it had forwarded the query message earlier.

Then R_{10} prepares a message and digest and sends it to its upstream RSU as follows: Message from R_{10} to R_6 : $\langle ENC_{PU_{R_6}}(T, SID, R_{10}, R_{11}) \rangle$ and digest $\langle h(T||SID||R_{10}||R_{11}) \rangle$ for integrity check. Other RSUs just discard the message and digest as they cannot decrypt the message.

After receiving the message, R_6 knows that it is a reply message for some query he sent earlier. To know more about the query, it decrypts it with its private key and checks its database to see if it matches the navigation credentials: $\langle NQ, SID \rangle$ along with the partial navigation route $\langle R_{10} \rangle$ to whom it had forwarded the query message earlier. Then R_6 prepares a message and digest and sends both to its upstream RSU as follows:

$$\begin{aligned} &\langle ENC_{PU_{R_5}}(T, SID, R_6, R_{10}, R_{11}) \rangle \\ &\langle h(T||SID||R_6||R_{10}||R_{11}) \rangle. \end{aligned}$$

Only R_5 can decrypt and then check its database for a match with navigation credentials $\langle NQ, SID \rangle$. On matching, it prepares a similar reply message and sends it to GLP along with the digest as follows:

$$\begin{aligned} &\langle ENC_{PU_{GLP}}(T, SID, R_5, R_6, R_{10}, R_{11}) \rangle \\ &\langle h(T||SID||R_5||R_6||R_{10}||R_{11}) \rangle \end{aligned}$$

GLP , after receiving the message and digest, checks its database if there is a match with the navigation credentials $\langle NQ, SID \rangle$. On matching, it retrieves the old session-ID (SND) from its database and prepares the final reply message containing the navigation route information \langle

Scyther results : verify				Claim	Status	Comments
IoVNavigation	RSU1	IoVNavigation,RSU1	Alive	Ok	Verified	No attacks.
		IoVNavigation,RSU2	Weakagree	Ok	Verified	No attacks.
		IoVNavigation,RSU3	Secret NQ1	Ok	Verified	No attacks.
		IoVNavigation,RSU4	Secret SID	Ok	Verified	No attacks.
		IoVNavigation,RSU5	Secret T1	Ok	Verified	No attacks.
		IoVNavigation,RSU6	Niagree	Ok	Verified	No attacks.
		IoVNavigation,RSU7	Nisynch	Ok	Verified	No attacks.
RSUj	IoVNavigation,RSUj1	IoVNavigation,RSUj1	Alive	Ok	Verified	No attacks.
		IoVNavigation,RSUj2	Weakagree	Ok	Verified	No attacks.
		IoVNavigation,RSUj3	Secret SID	Ok	Verified	No attacks.
		IoVNavigation,RSUj4	Secret T2	Ok	Verified	No attacks.
		IoVNavigation,RSUj5	Niagree	Ok	Verified	No attacks.
		IoVNavigation,RSUj6	Nisynch	Ok	Verified	No attacks.

Done.

Fig. 4. Scyther simulation result of our scheme.

$R_1, R_5, R_6, R_{10}, R_{11} \rangle$ for the query message NQ and the session-ID (SND) as follows:

$$\begin{aligned} NR &= R_1, R_5, R_6, R_{10}, R_{11} \\ NRE &= ENC_{PU_{V_i}}(SND, NQ, NR) \end{aligned}$$

Then sends the message $\langle PRID_1, PID_{V_i}, Cert_{GLP}, NRE \rangle$ along with the signature $SIG_{PR_{GLP}}(PRID_1, PID_{V_i}, SND)$.

D. RSU-to-Vehicle Communication

After receiving the reply message from the GLP, R_1 verifies the message and checks the session-ID. On successful verification, it forwards the encrypted content NRE to the queried vehicle V_i as follows:

$$RM = \langle PRID_1, PID_{V_i}, SND, NRE \rangle \text{ with signature } SIG_{PR_{R_1}}h(PRID_1, PID_{V_i}, SND, NRE).$$

After receiving the message and signature, the vehicle first matches SND and then decrypts NRE with its own private key and verifies NQ . If it matches, then extracts NR and finds the navigation route as $\langle R_1, R_5, R_6, R_{10}, R_{11} \rangle$.

V. SECURITY VERIFICATION AND ANALYSIS

A. Formal Security Verification Using Scyther Tool

Automatic verification and simulation of our scheme is performed using the Scyther tool. It is a versatile tool for various security protocols under the perfect cryptography assumption [21]. The primary assumption is that the adversary can't infer anything from an encrypted message unless he is in possession of the decryption key. The simulation was performed by selecting the bounded and unbounded number of sessions. During simulation, the matching type is selected to check all types of attacks. The secret parameters, private keys, and identities of the entities have been tested against adversary attacks. Fig. 4 shows the simulation result of our protocol. This particular snapshot captures the simulation of our scheme between a pair of RSUs which is safe. The Nisynch feature (non-injective synchronization) ensures the

correct order of the messages without requiring unique identification of sessions. The Niagree (non-injective agreement) ensures that both parties agree on the completion and content of a session, even if sessions are not uniquely identified. The “Alive” feature ensures that the protocol terminates successfully and no party is stuck in the middle of the protocol. The “OK” and “verified” under the status column of the results section prove that the particular feature is supported by the protocol or that the particular parameter is protected against attacks.

B. Informal Security Analysis

Theorem 1: The proposed protocol preserves the integrity of the message and response.

Proof 1: In each stage, our scheme uses either the digital signature (ElGamal) or the one-way hash function (SHA-256) to verify the integrity of the messages and responses, respectively. In addition, applying the hash function before signing a message helps prevent the existential forgery attack.

Theorem 2: The proposed protocol provides anonymity between sender and receiver.

Proof 2: In our protocol, a sender (vehicle or RSU) uses the pseudo-identity to communicate with a receiver (vehicle or RSU). Since real identities are hidden from each other, anonymity is maintained during communication. The TA will reveal the real identity in case of any issues related to the behaviour of the sender or receiver.

Theorem 3: Provides unlinkability of the message/response with the sender or receiver.

Proof 3: Vehicles send navigation requests to the RSUs with their pseudonymous identities. In our scheme, the RSU that receives the query cannot see the navigation information, as it is encrypted with the public key of the GLP. Moreover, the other RSUs also pass the query from GLP to upstream/downstream RSUs with their respective pseudonyms. Since each of the query/response messages is accompanied by timestamps valid for a particular time window, neither linking an older message to its sender/receiver is possible nor finding the place/route of a specific navigation.

Theorem 4: The proposed protocol protects from desynchronization attacks.

Proof 4: In an authentication protocol, a desynchronization attack can occur if the protocol requires parameter updates in each session after execution. A mismatch in these updates could lead to such an attack. However, in our protocol, no such parameters are updated during the execution of the protocol. Therefore, desynchronization attacks are not possible.

Theorem 5: The proposed protocol is secure against replay attack.

Proof 5: Each navigation query and response are assigned with a unique session-ID and timestamp. This helps to avoid duplicate transmission in the system since both parameters are ephemeral and last for a very short time. So, our scheme is secure against replay attacks.

Theorem 6: The proposed protocol prevents an impersonation attack.

Proof 6: Registration is mandatory for the vehicle/RSU to participate in the navigation process. Without a valid pseudonym and session-ID, no entity can generate a valid navigation query/response message. Moreover, the TA issues public-key certificates only to registered entities. Since a pretender cannot avail a valid pseudonym, session-ID or certificate, it is immune to impersonation attacks.

Theorem 7: The proposed protocol is secure against the man-in-the-middle (MITM) attack.

Proof 7: Our protocol prevents MITM attacks since the messages are transmitted with their respective public key certificates (issued by the TA) and the digital signatures of the senders (respective private keys issued by the TA). Therefore, an attacker in the middle cannot hijack the communication, as it has no valid private/public key pair issued by the TA. However, if a person misuses his key pair, he could succeed in this attack. But an alert TA can catch the offender and revoke his certificate to drive him out of the network.

Theorem 8: The proposed protocol provides security against collusion attacks.

Proof 8: Our scheme ensures conditional protection against collusion attacks. According to the scheme, the first RSU that receives the navigation query from the vehicle is unknown about the start and end point of a navigation query. Moreover, other RSUs, even if come to know about the start and finish point, but they do not know the vehicle that sent the query. Since GLP acts as a trusted agent who is not supposed to collude with the RSUs, our scheme can provide protection from collusion attacks by the RSUs.

Theorem 9: The proposed protocol provides security against the Sybil attack.

Proof 9: Each entity must be registered with the TA before participating in vehicular communication. The TA then issues pseudonymous identities to maintain their anonymity. The private/public key pairs of each vehicle are associated with these pseudonyms. If an attacker creates fake identities, it needs to use the pseudonyms of others. Although the pseudonyms are part of the beacon signals transmitted by a vehicle/RSU, the private/public key pair is a part of the tamper-proof devices of the vehicle/RSU (also known as hardware security module) and hence hard to retrieve. Therefore, our scheme is resistant to Sybil attacks.

VI. PERFORMANCE ANALYSIS

Let T_{sig} and T_{ver} denote the cost of signing and verifying a message respectively. Similarly, the cost of an encryption, decryption, and a hash operation are denoted by T_{enc} , T_{dec} , and T_{hash} respectively. We refer to [22] for the computation cost of basic primitives on a platform like Ubuntu 18.04.4 LTS, memory: 7.7 GiB, processor: IntelCore i7-8565U @ 1.80 GHz \times 8 and disk: 966.1 GB. The average computational cost has been reported as follows:

- 1) modular exponentiation = $T_{exp} = 0.072$ ms
- 2) modular multiplication = $T_{mul} = 0.002$ ms
- 3) hash operation = $T_{hash} = 0.055$ ms.

TABLE II
PERFORMANCE COMPARISON

Features	VSPN[11]	SPNS[13]	PiSim[14]	EPNS[16]	Ours
Computation Cost	High	High	High	Low	Low
Cryptography	ECC, Bilinear Pairing	AES, ElGamal, Bilinear Pairing	Bilinear Pairing	Secure Multiparty Computation	ElGamal
Group signature	Yes	Yes	Yes	No	No
Crowdsourcing	No	Yes	Yes	No	No
Fog computing	No	Yes	No	No	No
Message Forwarding	point-to-point	broadcast	broadcast	broadcast	point-to-point
Network Traffic	Low	High	High	High	Low

TABLE III
COMPUTATION COST (IN MS)

Navigation Query	Navigation Response	Total (in ms)
$2.913 + 0.902n$	$1.247 + 0.471n$	$4.160 + 1.373n$

TABLE IV
COMMUNICATION COST (IN BYTES)

Navigation Query	Navigation Response	Total (in bytes)
$4252 + 512n$	$2212 + 288n$	$6464 + 800n$

Since our scheme uses El-Gamal encryption and signature schemes, so $T_{enc} = 2T_{exp} + 1T_{mul} = 0.146$ ms and $T_{dec} = 2T_{exp} + 1T_{mul} + 1T_{add} = 0.147$ ms. Similarly $T_{sig} = 2T_{exp} + 2T_{mul} + 1T_{hash} + 1T_{add} = 0.204$ ms and $T_{ver} = 3T_{exp} + 1T_{mul} + 1T_{hash} = 0.273$ ms.

The computation cost of a query from the vehicle to the first RSU is $T_{sig} + 2T_{ver} = 0.750$ ms. Similarly, the cost of a response from the RSU to the queried vehicle is $T_{enc} + T_{sig} + T_{ver} + T_{dec} = 0.770$ ms. The vehicle then sends the final message to the RSU with a cost of $T_{enc} + T_{sig} = 0.350$ ms. Then the RSU verifies the signature and sends a message to the GLP that costs $T_{ver} + T_{enc} + T_{sig} = 0.623$ ms. Finally, the computation cost at GLP is $T_{ver} + T_{dec} = 0.420$ ms.

Subsequently, the communication between RSU to RSU takes place hop-by-hop. So each of the RSU first decrypts a message received by it and then verify the signature which costs $T_{dec} + T_{ver} = 0.497$ ms. Then it prepares a message and signature which costs $T_{enc} + T_{sig} + T_{hash} = 0.405$ ms.

Note that if there are n such RSUs from the source RSU to the destination RSU, then the total intermediate RSU computation cost would be $n \times (0.497 + 0.405) = 0.902n$.

For the navigation reply, each RSU prepares an encrypted message with a digital digest which costs $T_{enc} + T_{hash} = 0.471$ ms. This would cost $0.471n$ for n intermediate RSUs as above. The GLP then sends the reply to the first RSU with a cost of $T_{enc} + T_{sig} = 0.350$ ms.

Finally, the cost of computation incurred between the first RSU and the requested vehicle would be $2T_{ver} + T_{sig} + T_{dec} = 0.897$ ms.

Table II presents a comparative overview of our navigation scheme with some of the similar schemes carried out by researchers in recent times. Since there is limited scope for a direct comparison of computation and communication costs of these works (since their approaches are different), we only highlight a few relevant features to get an overall idea of their performance. Tables III and IV present the computation and communication cost of our proposed scheme, respectively.

VII. CONCLUSION

In this paper, we have proposed a secure and dynamic route navigation and authentication scheme using the capability of RSUs in IoV for smart city vehicular communication. It begins with the static route provided by the GLP. Then it uses a chain of RSUs to discover a secure and dynamic route from a source to a destination with minimal traffic load and congestion. Therefore, our scheme searches for a sub-optimal route with a decentralized approach without directly depending on the TA or the vehicles. GLP, in our scheme, plays a significant role in isolating the vehicle/RSU that requested the navigation query from the set of RSUs who discovered the route. The intuition behind this idea is to achieve driver anonymity and protection against collusion attacks. We have also verified our scheme against other possible attacks and have found it safe to deploy in vehicular communication. Our scheme uses pseudonym-based asymmetric key cryptography for encryption/decryption and digital signature under the ElGamal cryptosystem. The computation overhead is also minimal on the vehicles and the RSUs, since it does not use ECC (Elliptic Curve Cryptography) or PBC (Pairing-based cryptography). refine and make it short

REFERENCES

- [1] F. Yang, S. Wang, J. Li, Z. Liu, and Q. Sun, "An overview of internet of vehicles," *China Commun.*, vol. 11, no. 10, pp. 1–15, 2014.
- [2] A. Rahim et al., "Vehicular social networks: A survey," *Pervasive Mobile Comput.*, vol. 43, pp. 96–113, 2018.
- [3] A. M. Vegni and V. Loscri, "A survey on vehicular social networks," *IEEE Commun. Surv. Tut.*, vol. 17, no. 4, pp. 2397–2419, Fourthquarter 2015.
- [4] Z. Ning, F. Xia, N. Ullah, X. Kong, and X. Hu, "Vehicular social networks: Enabling smart mobility," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 16–55, May 2017.
- [5] S. K. Dwivedi, R. Amin, S. Vollala, and M. K. Khan, "B-HAS: Blockchain-assisted efficient handover authentication and secure communication protocol in VANETs," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 6, pp. 3491–3504, Nov./Dec. 2023.

- [6] H. Zhong, L. Chen, J. Cui, J. Zhang, I. Bolodurina, and L. Liu, "Secure and lightweight conditional privacy-preserving authentication for fog-based vehicular ad hoc networks," *IEEE Internet Things J.*, vol. 9, no. 11, pp. 8485–8497, Jun. 2022.
- [7] M. Wazid, P. Bagga, A. K. Das, S. Shetty, J. J. P. C. Rodrigues, and Y. Park, "Akm-iov: Authenticated key management protocol in fog computing-based internet of vehicles deployment," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8804–8817, Oct. 2019.
- [8] K. Zhang, J. Ni, K. Yang, X. Liang, J. Ren, and X. S. Shen, "Security and privacy in smart city applications: Challenges and solutions," *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 122–129, Jan. 2017.
- [9] J. Petit, F. Schaub, M. Feiri, and F. Kargl, "Pseudonym schemes in vehicular networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 228–255, Firstquarter 2015.
- [10] H. Jin, M. Khodaei, and P. Papadimitratos, "Security and privacy in vehicular social networks," in *Vehicular Social Networks*, CRC Press, 2017, pp. 155–169.
- [11] T. Chim, S. Yiu, L. C. Hui, and V. O. K. Li, "VSPN: VANET-based secure and privacy-preserving navigation," *IEEE Trans. Comput.*, vol. 63, no. 2, pp. 510–524, Feb. 2014.
- [12] J. Ni, X. Lin, K. Zhang, and X. Shen, "Privacy-preserving real-time navigation system using vehicular crowdsourcing," in *Proc. 2016 IEEE 84th Veh. Technol. Conf.*, 2016, pp. 1–5.
- [13] L. Wang, G. Liu, and L. Sun, "A secure and privacy-preserving navigation scheme using spatial crowdsourcing in fog-based VANETs," *Sensors*, vol. 17, no. 4, 2017, Art. no. 668. [Online]. Available: <https://www.mdpi.com/1424-8220/17/4/668>
- [14] M. Li, Y. Chen, S. Zheng, D. Hu, C. Lal, and M. Conti, "Privacy-preserving navigation supporting similar queries in vehicular networks," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 2, pp. 1133–1148, Mar./Apr. 2022.
- [15] J. Ni, K. Zhang, Y. Yu, X. Lin, and X. Shen, "Privacy-preserving smart parking navigation supporting efficient driving guidance retrieval," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6504–6517, Jul. 2018.
- [16] J. Zhou, S. Chen, K.-K. R. Choo, Z. Cao, and X. Dong, "EPNS: Efficient privacy-preserving intelligent traffic navigation from multiparty delegated computation in cloud-assisted VANETs," *IEEE Trans. Mobile Comput.*, vol. 22, no. 3, pp. 1491–1506, Mar. 2023.
- [17] B. Baruah and S. Dhal, "A security and privacy preserved intelligent vehicle navigation system," *IEEE Trans. Dependable Secure Comput.*, vol. 20, no. 2, pp. 944–959, Mar./Apr. 2023.
- [18] J. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *IEEE Secur. Privacy*, vol. 2, no. 3, pp. 49–55, May/Jun. 2004.
- [19] D. Huang, S. Misra, M. Verma, and G. Xue, "PACP: An efficient pseudonymous authentication-based conditional privacy protocol for VANETs," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 736–746, Sep. 2011.
- [20] J. Kang, R. Yu, X. Huang, and Y. Zhang, "Privacy-preserved pseudonym scheme for fog computing supported internet of vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2627–2637, Aug. 2018.
- [21] C. J. F. Cremers, "The scyther tool: Verification, falsification, and analysis of security protocols," in *Computer Aided Verification*, A. Gupta and S. Malik, Eds. Berlin, Germany: Springer, 2008, pp. 414–418.
- [22] P. Bagga, A. K. Sutrala, A. K. Das, and P. Vijayakumar, "Blockchain-based batch authentication protocol for internet of vehicles," *J. Syst. Archit.*, vol. 113, 2021, Art. no. 101877. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1383762120301569>



Bimal Kumar Meher (Member, IEEE) received the M.Sc. degree in electronics from Berhampur University, Berhampur, India, in 1995, and the M.Tech (with Hons.) and Ph.D. degrees in computer science from Utkal University, Bhubaneswar, India, in 2002 and 2016, respectively. He is currently an Associate Professor with the Department of Computer Science and Engineering, Silicon University, Bhubaneswar, Odisha, India. His research interests mainly include algorithms and architecture design for finite field arithmetic, ECC-based protocol design, multi-factor authentication, WBAN and IoV security, and Blockchain. He was the recipient of the Sydney R. Parker Best Paper Award.



Ruhul Amin (Senior Member, IEEE) received the B.Tech. and M.Tech. degrees in computer science and engineering from the Maulana Abul Kalam Azad University of Technology, West Bengal, India, in 2009 and 2013, respectively, and the Doctoral (Ph.D.) degree in computer science and engineering from the Indian Institute of Technology (ISM) Dhanbad, Jharkhand, India, in 2017. He is currently an Assistant Professor with the Department of Computer Science and Engineering, IIIT Naya Raipur, Chhattisgarh, India. He has authored many technical research papers published in leading international conferences and peer reviewed international journals. His research interest includes cryptography and network security, authentication protocol, WSN security, and IoT security.



Mohammad Abdussami received the Doctoral (Ph.D.) degree in computer science and engineering from the International Institute of Information Technology, Naya Raipur (IIIT-NR), Chhattisgarh, India, in 2024. He is currently an Assistant Professor with SRM University AP, Amaravati, Andhra Pradesh, India. He has authored or coauthored research papers in renowned international journals and conferences. His research interests include lightweight cryptography, network security, and post quantum cryptography.



Muhammad Khurram Khan (Senior Member, IEEE) is currently a Professor of cybersecurity with the Center of Excellence in Information Assurance, King Saud University, Kingdom of Saudi Arabia. He is also the founder and CEO with the Global Foundation for Cyber Studies and Research, an independent and nonpartisan cybersecurity think-tank in Washington, D.C., USA. He is also on the Editorial Board of several journals including, *IEEE COMMUNICATIONS SURVEYS & TUTORIALS*, *IEEE Communications Magazine*, *IEEE INTERNET OF THINGS JOURNAL* and many more. His research interests include cybersecurity, digital authentication, IoT security, biometrics, cyber policy, and technological innovation management. He is a Distinguished Lecturer of the IEEE. He is also a Fellow of IET (U.K.), BCS (U.K.), and FTRA (Korea).



Md Abdul Saifulla received the Ph.D. degree in computer science and engineering from Anna University, Chennai, Tamil Nadu. He is currently an Assistant Professor with the School of Computer and Information Sciences, University of Hyderabad, Hyderabad, Telangana, India. He has authored or coauthored multiple papers in peer reviewed international journals (IEEE, Elsevier, and Springer) and presented multiple papers in reputed international national conferences. His research interests include network traffic analysis, data center products, network management, software defined networking, named data networking, and quantum communications. He was the recipient of the Gold Medal for academic excellence from MEDu Society, Hyderabad.



Sanjeev Kumar Dwivedi received the Doctoral (Ph.D.) degree in computer science and engineering from the International Institute of Information Technology, Naya Raipur (IIIT-NR), Chhattisgarh, India, in 2023. He is currently an Assistant Professor with Artificial Intelligence and Data Science Programme, Centre for Artificial Intelligence, Madhav Institute of Technology & Science, Deemed University (MITS-DU), Gwalior, Madhya Pradesh, India. He has authored or coauthored research papers in renowned international journals and conferences. His research interests include authentication protocols, cryptography, blockchain applications, digital-twins, and post quantum cryptography.