TrafficProof: Privacy-Preserving Reliable Traffic Information Sharing in Social Internet of Vehicles

Stefan Dziembowski*, Shahriar Ebrahimi†, Parisa Hassanizadeh†‡ and Susil Kumar Mohanty*§
*University of Warsaw, Poland. Emails: stefan.dziembowski@crypto.edu.pl, s.mohanty@uw.edu.pl
† Zero Savvy, United Kingdom. Email: {sh.ebrahimi92}, {parisaa.hassanizadeh} @gmail.com
‡ Institute of Fundamental Technological Research, Polish Academy of Science (IPPT PAN), Poland.
§ Corresponding Author

***Abstract*—In the Social Internet of Vehicles (SIoV), effective data sharing is essential for applications including road safety, traffic management, and situational awareness. However, the decentralized and open nature of SIoV presents significant challenges in simultaneously ensuring data integrity, user privacy, and system accountability. This paper presents a protocol for secure and location-accurate traffic data sharing that fully preserves the anonymity and privacy of participating witnesses. The protocol leverages zero-knowledge proofs (ZKPs) to allow vehicles to broadcast redacted traffic information—such as images—tied to specific geographic locations, while withholding both the original content and the identity of the reporting vehicle. To ensure the authenticity of the redacted content and the legitimacy of the witness, an additional ZKP is used to privately validate both elements. Upon receiving a report, the verifying node checks the submitted proofs, aggregates validated inputs, and publishes the resulting metadata to both IPFS and a blockchain. This design ensures public verifiability, tamper resistance, and the reliability of the shared data, while maintaining strong privacy guarantees through cryptographic anonymity. To improve the efficiency of proof generation on resource-constrained devices, the protocol employs folding-based ZKP constructions. We conduct a formal security and soundness analysis of the protocol and implement a proof-of-concept, which is publicly available as open-source software. Experimental evaluations on commodity hardware demonstrate that the protocol is computationally efficient and introduces less than 1.5% communication overhead relative to the size of the shared traffic data, indicating its suitability for real-world deployment.**

***Index Terms*—Data Authenticity, Proof of Location, Reliability, zkSNARKs, Social Internet of Vehicles**

1. Introduction

The Social Internet of Vehicles (SIoV) [1], [2], [3], [4] is a rapidly evolving paradigm that transforms vehicles into intelligent, cooperative agents capable of real-time data exchange with their environment. Unlike traditional vehicular networks where vehicles operate independently, SIoV enables vehicles to communicate and collaborate with other vehicles (V2V), roadside infrastructure (V2I), mobile devices and cloud services (V2N), pedestrians (V2P), blockchains (V2B). This interconnected system allows vehicles to share crucial information in real-time, including: safety-critical events (e.g., accidents and collisions, road hazards, sudden braking, emergency vehicle proximity) [5]; traffic flow and congestion (e.g., traffic jams, detours, signal failures) [6]; environmental and road conditions (e.g., adverse weather, pollution alerts, flooding) [7]; parking and infrastructure updates (e.g., parking availability, toll queue lengths, charging station status) [8]; and social or informational broadcasts (e.g., public announcements, commercial messages) [9]. These services significantly contribute to smarter transportation systems, offering not only improved road safety, traffic flow, travel efficiency, and safe driving experience but also ecological and economic benefits by minimizing congestion and reducing fuel consumption.

1.1. Challenges of Data Sharing in SIoV

As the SIoV becomes an integral component of smart city infrastructure, its reliance on vehicle-generated real-time data introduces a new set of critical challenges. Foremost among these is the threat posed by the dissemination of false, misleading, or fabricated information. In critical scenarios such as accident alerts, road hazards, or traffic congestion reports, even a single inaccurate message can lead to disruptive traffic decisions, misinformed rerouting, or in extreme cases, life-threatening outcomes. Therefore, guaranteeing the authenticity and trustworthiness of shared data is not a mere design consideration, it is a vital requirement. Although OnionChain [10] initially acknowledges this challenge, it explicitly considers the detection of false data to be beyond the scope of their work. To mitigate this challenge, various prior studies have put forth promising approaches. Such as, SIoVChain [11] incentivizes vehicles to share accurate information by offering rewards for correct data and imposing penalties for false messages. MuSigRDT [12] adopts a threshold Schnorr multisignature-based approach, aggregating corroborative evidence from multiple independent vehicles before treating information as reliable. Similarly, CreditCoin [9] utilizes a consensus-driven validation mechanism to enhance data integrity and

reduce the influence of deceptive messages. Nevertheless, despite these advancements, none of the existing solutions can definitively determine whether the shared information is authentic and trustworthy. **RQ**[1]: How can a vehicle reliably prove the authenticity of data and its location to a third party without exposing its identity or sensitive information, while ensuring trust for coordinated actions such as rerouting or accident reporting? To the best of our knowledge, this work is the first to propose a verifiable framework that directly addresses the challenge of proving the correctness of shared information in SIoV.

Another major concern arises when a vehicle shares information that is factually correct, but not relevant to its current location. In such cases, it could pertain to a different area or traffic context, leading to potential misinterpretation or misuse. Therefore, it becomes essential to not only validate the content of the information but also to prove that the vehicle was physically present at the claimed location, known as proof of location.

Several prior works on privacy-preserving location-based services [3], [4], [13], [14], [15] primarily focus on protecting the location privacy of responders (e.g., RSUs) from potential verifiers. In contrast, our work addresses the inverse challenge: enabling a witness vehicle to publicly disclose its location in order to prove its physical presence at a specific point in space, while maintaining complete anonymity with respect to its identity. In this model, the vehicle provides cryptographic evidence of its location to a responder (e.g., an RSU) without revealing any identifying information or depending on a trusted third party. Despite the transparency of the location data, the use of ZKPs ensures that the witness's identity remains confidential. This approach enables a trustless, verifiable, and privacy-preserving data-sharing process that delivers a strong guarantee of reliability and authenticity—properties that are often not jointly achieved in existing solutions. **RQ:** How can a vehicle anonymously prove its physical presence at a claimed location to a responder, while still enabling trust-based decision-making for critical applications such as emergency response?

A further challenge arises when the data shared by a vehicle is valid in terms of authenticity, location relevance, and proof of presence, but is temporally outdated, i.e., it falls outside an acceptable time window. In dynamic traffic environments, even accurate and location-valid data can be misleading if not timely. Thus, it is also crucial to ensure that the information is not only genuine and location-bound but also generated and shared within a specific, recent timeframe to maintain its effectiveness for real-time decision-making in SIoV [6]. **RQ:** How can we ensure that shared data is generated within a valid time window to support accurate and timely decision-making in SIoV?

Furthermore, user privacy and anonymity remain critical concerns in SIoV [16]. Data shared by vehicles may unintentionally expose sensitive information, such as registration numbers, driver or owner identities, or facial images captured through dashcams and surveillance systems. Without proper anonymization and protection, such data can lead to privacy breaches, tracking, profiling, or even identity theft. Therefore, it is essential to incorporate robust mechanisms that preserve both data privacy and user anonymity in SIoV. Existing privacy-preserving solutions [1], [9], [14], [17], [18], [19] are often computationally and communication-intensive, primarily due to their reliance on heavyweight cryptographic primitives such as group signatures, threshold ring signatures, bilinear pairings, attribute-based encryption, and partially homomorphic encryption. **RQ:** How can we design a highly efficient and lightweight solution, achieving minimal computation (within a few milliseconds) and low communication overhead (a few kilobytes) that enables anonymous data sharing in SIoV while preserving the privacy of sensitive information and behavioral patterns?

1.2. Our Approach

Findings from [16] indicate that many participants consider the safety benefits of data sharing to outweigh potential privacy concerns. This implies that, as long as strong privacy protections are in place, users are likely to be cooperative in sharing their data—making privacy-preserving data collection both feasible and practical. Motivated by these findings, we propose a method designed to address the aforementioned challenges while preserving user privacy and ensuring data utility.

We present a privacy-preserving and verifiable data-sharing architecture for SIoV systems. In the proposed protocol, vehicles assume one of two roles: *requesters*, which issue queries about the status of a specific location, and *witnesses*, which provide relevant information regarding that location. Witness vehicles generate zero-knowledge proofs (ZKPs) to demonstrate the validity of their shared data while maintaining anonymity, including from Roadside Units (RSUs). The protocol is designed to support a decentralized and trustless environment in which location-specific updates are publicly verifiable yet do not compromise the privacy of contributing vehicles.

To coordinate data requests and location state updates, a smart contract manages on-chain metadata, including content identifiers (CIDs) corresponding to shared information. The underlying data—such as redacted images and their associated proofs—is stored off-chain using the InterPlanetary File System (IPFS), ensuring both data integrity and efficient storage.

At the core of the protocol is an efficient proof-of-redaction mechanism tailored for traffic data in compressed JPEG format. This design choice significantly reduces computational overhead compared to approaches that operate on uncompressed data. To preserve privacy during presence authentication (proof of location), we employ a lightweight hash-based commitment scheme that conceals vehicle identities from RSUs. This allows vehicles to credibly assert claims—such as witnessing an event at a specific location—without revealing their identity, while being able to authenticate their redacted data.

1. RQ: Research question

This work aims to balance transparency and confidentiality by enabling a secure, decentralized, and trustworthy SIoV ecosystem that supports incentive-compatible participation, preserves user privacy, and ensures the integrity of safety-critical information. We conduct a comprehensive security and soundness analysis of the proposed protocol and implement a proof-of-concept system, which is made available as an open-source repository on GitHub.

The implementation is evaluated on commodity hardware (e.g., a standard laptop) to assess its computational and communication overhead. Experimental results show that the system imposes minimal performance costs. In particular, generating a proof of redaction for a high-resolution image requires approximately 23 seconds, and the total size of the proof and associated public values does not exceed 14 KB. This constitutes roughly 1.5% of the original image size, demonstrating the efficiency and practicality of the framework for deployment in real-world vehicular networks.

The key contributions of this paper are as follows:

Anonymous but Reliable Traffic Data Sharing The proposed framework enables vehicles to share safety-critical information (e.g., accidents, road hazards, traffic conditions) in a publicly verifiable and trustless manner. To achieve this, a vehicle generates ZKPs that attest to the authenticity of redacted data while preserving anonymity. Additionally, a group of authenticated RSUs sign specific commitments to the vehicle's data, attesting to its presence at a given location and time without revealing the vehicle's identity.

Proof of Redaction We introduce an efficient proof-of-redaction mechanism for images in the compressed JPEG format. This approach significantly reduces the prover's computational complexity compared to existing techniques that operate on uncompressed (pixel-level) data. To ensure the integrity of redactions, we propose a block-based hashing mechanism that operates directly on the compressed image data.

Proof of Concept We implement the proposed protocol as an open-source repository on GitHub. Our proof-of-concept implementation is evaluated on commodity hardware, and we analyze its performance in terms of both computational and communication overhead.

Practical Overhead Experimental results indicate that the system incurs low computational and communication overhead, demonstrating its practicality for deployment in real-world vehicular networks.

Security and Robustness Analysis We present a comprehensive security and soundness analysis of the proposed protocol.

The remainder of the paper is organized as follows. Section 2 introduces the application of the proposed protocol. Section 3 reviews the cryptographic primitives used throughout the protocol. Section 4 outlines the system architecture and adversary models. Section 5 describes the components of the proposed protocol in detail. Section 6 presents the implementation results. Section 7 compares our approach with related work, and Section 8 concludes the paper.

2. Applications

This section highlights several potential applications of the proposed framework within the broader landscape of the Social Internet of Vehicles (SIoV). Although the discussion is presented informally, our focus is on use cases that emphasize data authenticity, time-bounded relevance, and location specificity. We specifically examine traffic management, announcement dissemination, and road safety as key representative domains, offering a detailed analysis of how the framework can support timely and trustworthy information sharing in these contexts. While our analysis centers on these particular applications, we believe the underlying principles are widely applicable across a variety of SIoV use cases. The extension of this framework to other domains is left as an open direction for future research.

2.1. Traffic Management

Traffic management aims to enhance the efficiency and fluidity of vehicle flow by utilizing aggregated data on speed, density, and route preferences. It typically involves periodic communication with centralized or distributed control systems that dynamically optimize traffic signals and routing paths. Unlike safety-focused scenarios, the emphasis here is on data integrity and system-wide coordination rather than individual user privacy. While scalability and interoperability with heterogeneous infrastructure are ongoing challenges, ensuring that traffic data is accurate and resistant to manipulation is critical to avoiding congestion and rerouting failures.

2.2. Announcement Dissemination

Announcement dissemination allows vehicles to broadcast situational messages, such as accidents, congestion, or weather hazards to others in proximity or along a planned route. These messages are often initiated spontaneously and may travel across multiple hops to reach relevant users. The key challenge lies in enabling anonymous but verifiable information sharing, often requiring proof-of-location or digital signatures to establish message authenticity without exposing user identity. Additionally, mechanisms are needed to filter redundant or malicious data and to incentivize honest participation. This scenario is uniquely positioned between safety and social information exchange, demanding contextual relevance and location-aware dissemination.

2.3. Road Safety Management

Road safety management in SIoV is fundamentally centered on preserving human life by preventing accidents and hazardous interactions on the road. It operates through real-time exchange of critical event-driven data, such as sudden braking, obstacle detection, or erratic driving primarily via V2V and V2I channels. This application demands ultra-reliable and low-latency communication to ensure that warnings are received in time to mitigate risk. While privacy

TABLE 1. TERMINOLOGY OF THE PAPER

Notation	Description
α, β	We refer to the data of the *original* and the *redacted* image as α and β, respectively.
α_i, β_i	i-th block of images α and β.
H	Poseidon [20] hash function. $H : \mathbb{Z}_p^2 \to \mathbb{Z}_p$
H_ϕ	Hash value of an entire image with $n \times m$ blocks: $H_\phi : \mathbb{Z}_p^{n \times m} \to \mathbb{Z}_p = h^n$.
f_R	Redaction function: $\beta \leftarrow f_R(\alpha, M)$
M	masking map appliable on original image for redaction
$\mathcal{V}$	Verifier algorithm, we use $\mathcal{V}_\sigma$, $\mathcal{V}_\pi$, and $\mathcal{V}_\Delta$ as signature, zkSNARKs proof, and Merkle proof verifiers.
π	we refer to zkSNARKs proofs as π. We specifically use π_β, π_Δ, π_σ for proofs of image redaction, Merkle inclusion, and signature verification proofs.
σ_m	Digital signature of message m.
Δ_{PK}	Root of the Merkle tree built over the public keys.
δ_i	Merkle path of the i^{th} leaf.
$\mathcal{R}_{eq}$	Requester vehicle
$\mathcal{R}_{es}$	Responder RSU
$\mathcal{W}$	Witness vehicle
$\mathcal{L}$	Location
η, r	random numbers

is maintained under normal conditions through anonymous messaging, conditional traceability is essential to identify malicious behavior post-incident. Thus, safety systems must balance data trustworthiness with user anonymity in high-risk, time-sensitive environments.

2.4. Platooning Service

Platooning involves tightly coordinated driving among a group of vehicles, typically led by a head vehicle, to reduce fuel consumption, increase road utilization, and improve safety. This service relies on persistent, low-latency V2V communication to maintain synchronized movement, lane changes, and braking. Security in platooning revolves around ensuring only trusted participants join the group, protecting control messages from tampering, and maintaining operational privacy, particularly regarding vehicle identity and location over time. Given the cooperative and closed nature of this service, identity privacy is often preserved within the platoon, while message integrity and access control are prioritized to avoid infiltration or command spoofing.

3. Background

This section provides the necessary background for understanding the remainder of the paper. We begin by introducing fundamental concepts related to non-interactive proof systems, with a particular focus on SNARKs. We then proceed to formally define the properties of Incrementally Verifiable Computation (IVC) schemes, which represent a specialized subclass of SNARKs. Table 1 summarizes the key symbols and terminology used throughout the paper.

Definition 1. ***([zk]SNARKs)*** *Let $\mathcal{R}$ be a binary relation for an NP language $L_\mathcal{R}$, where λ is the security parameter. The argument system for $\mathcal{R}$ is defined as a quadruple probabilistic polynomial algorithms $\Pi = (\mathcal{G}, \mathcal{P}, \mathcal{V}, \mathcal{S})$ and a deterministic encoder $\mathcal{K}$, where:*

- *$pp \leftarrow \mathcal{G}(1^\lambda)$: The generator samples the public parameter pp w.r.t. the security parameter λ.*
- *$(pk, vk) \leftarrow \mathcal{K}(pp, s)$: The prover and verifier key pair is derived from the commonly defined structure s and the public parameter pp using the deterministic encoder.*
- *$\pi \leftarrow \mathcal{P}(pk, u, w)$: Proving algorithm stating $(pp, s, u, w) \in \mathcal{R}$.*
- *$b \leftarrow \mathcal{V}(vk, u, \pi)$: Verification algorithm, where $b \in \{0, 1\}$.*
- *$\pi \leftarrow \mathcal{S}(pp, u, \tau)$: Simulator outputs π given trapdoor τ.*

We further call a zero-knowledge non-interactive argument for $\mathcal{R}$ as a zkSNARK if it satisfies:

- ***Completeness**: An honest prover with valid witness should convince any verifier. Formally, for any PPT adversary $\mathcal{A}$:*

$$Pr\left[\mathcal{V}(vk, u, \pi) = 1 \;\middle|\; \begin{array}{l} pp \leftarrow \mathcal{G}(1^\lambda) \\ (s, (u, w))) \leftarrow \mathcal{A}(pp) \\ (pp, s, u, w) \in \mathcal{R} \\ (pk, vk) \leftarrow \mathcal{K}(pp, s) \\ \pi \leftarrow \mathcal{P}(pk, u, w) \end{array}\right] = 1$$

- ***Knowledge Soundness**: A dishonest prover (adversary), should not be able to convince any verifier. To formally define this we require that for all PPT adversaries $\mathcal{A}$ there exists an extractor $\mathcal{E}$ that can compute witness given any randomness ρ, such that:*

$$Pr\left[\begin{array}{l} \mathcal{V}(vk, u, \pi) = 1, \\ (pp, s, u, w) \notin \mathcal{R} \end{array} \;\middle|\; \begin{array}{l} pp \leftarrow \mathcal{G}(1^\lambda) \\ (s, (u, w))) \leftarrow \mathcal{A}(pp) \\ (pk, vk) \leftarrow \mathcal{K}(pp, s) \\ w \leftarrow \mathcal{E}(pp, \rho) \end{array}\right] = negl(\lambda)$$

- ***Zero-knowledge**: If the argument does not reveal anything beyond the truth of the statement, we label it as zero-knowledge. Formally, there must exist a PPT simulator $\mathcal{S}$ such that for all PPT adversaries $\mathcal{A}$ following distributions are indistinguishable:*

$$\mathcal{D}_1 = \left\{ (pp, s, u, \pi) \;\middle|\; \begin{array}{l} pp \leftarrow \mathcal{G}(1^\lambda) \\ (s, (u, w))) \leftarrow \mathcal{A}(pp) \\ (pp, s, u, w) \in \mathcal{R} \\ (pk, vk) \leftarrow \mathcal{K}(pp, s) \\ \pi \leftarrow \mathcal{P}(pk, u, w) \end{array}\right\}$$

$$\approx$$

$$\mathcal{D}_2 = \left\{ (pp, s, u, \pi) \;\middle|\; \begin{array}{l} (pp, \rho) \leftarrow \mathcal{S}(1^\lambda) \\ (s, (u, w))) \leftarrow \mathcal{A}(pp) \\ (pp, s, u, w) \in \mathcal{R} \\ (pk, vk) \leftarrow \mathcal{K}(pp, s) \\ \pi \leftarrow \mathcal{S}(pp, u, \rho) \end{array}\right\}$$

Definition 2. ***Folding-based zkSNARKs:*** *Folding schemes achieve efficient incrementally verifiable computation (IVC) and enable asserting computations that involve repeated applications of the same function [21]. Formally, given a function f and an initial input z_0, the goal is to verify that $z_i = f^i(z_0)$ for some iteration i. Folding schemes achieve*

this by allowing the generation of a proof Π_i asserting the correctness of z_i, assuming the validity of a prior proof Π_{i-1} attesting that $z_{i-1} = f^{i-1}(z_0)$. A notable feature of these schemes is their ability to accommodate auxiliary inputs. While the primary input to f at each step is derived from the previous output, an additional auxiliary input ω_i can be provided independently at each iteration. This capability allows IVC to generalize traditional SNARK completeness and soundness guarantees to settings where each application of f may depend on step-specific external data.

We formally define IVC by PPT algorithms $(\mathcal{G}, \mathcal{P}, \mathcal{V})$ and deterministic encoder $\mathcal{K}$ satisfying:

- ***Completeness**: For any PPT adversary $\mathcal{A}$:*

$$Pr\left[\begin{array}{l|l} \mathcal{V}(vk, i, z_0, \\ z_i, \Pi_i) = 1 \end{array} \middle| \begin{array}{l} pp \leftarrow \mathcal{G}(1^\lambda) \\ f, (i, z_0, z_{i-1}, z_i, w_{i-1}, \Pi_{i-1}) \leftarrow \mathcal{A}(pp) \\ z_i = f(z_{i-1}, \omega_{i-1}) \\ (pk, vk) \leftarrow \mathcal{K}(pp, f) \\ \mathcal{V}(vk, i-1, z_0, z_{i-1}, \Pi_{i-1}) = 1 \\ \Pi_i \leftarrow \mathcal{P}(pk, i, z_0, z_i; z_{i-1}, \omega_{i-1}, \Pi_{i-1}) \end{array}\right] = 1$$

- ***Knowledge Soundness**: $\forall n \in \mathbb{N}$, and expected polynomial time adversaries $\mathcal{P}^*$, there exists expected polynomial time extractor $\mathcal{E}$, such for any randomness ρ, following probability is negligible:*

$$Pr\left[\begin{array}{l|l} z_n \neq z, \\ \mathcal{V}(vk, n, z_0, z, \Pi) = 1 \end{array} \middle| \begin{array}{l} pp \leftarrow \mathcal{G}(1^\lambda) \\ f, (z_0, z, \Pi) \leftarrow \mathcal{P}^*(pp; \rho) \\ (pk, vk) \leftarrow \mathcal{K}(pp, f) \\ (\omega_0, \ldots, \omega_{n-1}) \leftarrow \mathcal{E}(pp, z_0, z; \rho) \\ z_i \leftarrow f(z_{i-1}, \omega_{i-1}) \ \forall i \in \{1, \ldots, n\} \end{array}\right]$$

4. System and Adversary Model

In this section, we first describe the system model underlying the protocol. We then detail the adversary model, and finally, we outline the security and privacy goals of the protocol.

4.1. System Model

The system model includes five key components, detailed below: Certificate Authority, RSUs, Vehicles, Blockchain and Smart Contracts, and IPFS.

1) **Certificate Authority ($\mathcal{CA}$):** is responsible for issuing pseudonymous certificates to both vehicles and Roadside Units (RSUs), enabling secure and authenticated communication while preserving the anonymity of participating entities. Each certificate contains a public-private key pair along with a digital signature to ensure its authenticity. The $\mathcal{CA}$ maintains a confidential mapping between real identities and their corresponding pseudonyms, which may be revealed only under justified circumstances, such as dispute resolution. To ensure verifiability and integrity, the $\mathcal{CA}$ constructs a Merkle tree over pairs of generated public keys and associated random values, denoted as (pk, η), and commits to the resulting Merkle root. This Merkle root serves as a compact and tamper-evident representation of all issued credentials. Furthermore, the $\mathcal{CA}$ is responsible for maintaining and updating the Merkle structure over time by integrating new roots into a cumulative hash structure, thereby enabling efficient and secure updates of credential sets [22], [23].
2) **Roadside Units (RSUs):** RSUs are strategically positioned along roadways and serve as essential intermediaries in the communication ecosystem of the Social Internet of Vehicles. They enable seamless interaction across multiple communication modes, including Vehicle-to-Infrastructure (V2I), Vehicle-to-Blockchain (V2B), and Vehicle-to-Vehicle (V2V). Beyond facilitating data exchange, RSUs are entrusted with maintaining system integrity and availability. To ensure they remain trustworthy and uncompromised, RSUs periodically undergo remote attestation, a security procedure in which they generate cryptographic responses to publicly issued challenges, thereby proving their integrity and operational correctness within the network [24].
   - **Responder ($\mathcal{R}_{\text{es}}$):** Upon receiving a query, the responder RSU evaluates the request and the offered incentive. If the conditions are acceptable, the responder agrees to fulfill the request by providing the required information or service. To ensure accountability, the responder deposits a collateral amount as a commitment to delivering valid and accurate data.
   - **Signer ($\mathcal{R}_{\text{sign}}$):** These RSUs participate in the generation of location proofs by signing messages for the witness vehicles.
3) **Vehicle**: Each vehicle is equipped with an On-Board Unit (OBU) that enables wireless communication with other vehicles and RSUs. To ensure the confidentiality and integrity of sensitive data, every vehicle also incorporates a Tamper-Proof Device (TPD), such as a Trusted Platform Module (TPM) [25] that securely stores critical assets like the vehicle's private key. Based on their functional roles within the system, vehicles are classified into three distinct categories:
   - **Requester ($\mathcal{R}_{\text{eq}}$):** The requester is a vehicle seeking specific information related to a particular geographic area. This information may include road safety alerts, accident reports, location updates, driver or navigation assistance, collision warnings, traffic statistics, or toll payment data. In exchange for valid responses, the requester offers a predefined incentive.
   - **Witness ($\mathcal{W}$):** A witness is a nearby vehicle that supports the response process by providing verifiable evidence related to the queried information. If the witness's contribution is confirmed to be accurate, it receives a reward from the responder as an incentive for its participation and trust reinforcement.
4) **Blockchain and Smart Contract:** We adopt a general definition of blockchain, and the protocol is designed to be compatible with any underlying blockchain infrastructure.
5) **Inter Planetory File System (IPFS):** IPFS is a decentralized peer-to-peer storage network used to store encrypted raw data like traffic updates and sensor outputs. Each data item is identified by a cryptographic hash,

ensuring integrity and enabling efficient retrieval without needing to know its physical location.

4.2. Adversarial Model

We consider a semi-honest model in which RSUs are assumed to be honest-but-curious. That is, while they faithfully follow the prescribed protocol, they may attempt to infer private information about vehicles. RSUs are capable of broadcasting queries, collecting responses, and forwarding messages. An RSU may also act as a responder by providing requested information. Although RSUs may be vulnerable to compromise, such instances can be detected through periodic Remote Attestation (RA) procedures. A variety of RA protocols, such as the one proposed in [24], can be employed to assess the RSU's integrity and detect misbehavior. If a compromised RSU disseminates manipulated data, the protocol's built-in verification mechanisms can identify such deviations. In contrast, the participating vehicles namely, the requester $\mathcal{R}_{\text{eq}}$, responder $\mathcal{R}_{\text{es}}$, and witness $\mathcal{W}$ are assumed to be potentially malicious and mutually distrustful. A malicious requester $\mathcal{R}_{\text{eq}}$ may attempt to extract genuine information from the network without offering the promised incentive. Conversely, a dishonest responder $\mathcal{R}_{\text{es}}$ might seek to obtain the incentive from $\mathcal{R}_{\text{eq}}$ without providing valid or truthful information in return, and may also try to acquire witness signatures without offering compensation. Witness vehicles $\mathcal{W}$ are also susceptible to misbehavior. They are expected to act independently and may collude with other witnesses, the responder, or the requester. However, a dishonest witness may attempt to claim incentives from the responder without contributing accurate or verifiable support information.

The trust model adopted in our protocol is grounded in the fundamental assumption that the original image remains untampered. This assurance is established through a valid digital signature bound to the image, which is generated either by a tamper-proof capture device, such as the Sony Alpha 7 IV camera [26] or the Truepic Lens SDK for mobile devices [27] or by a trusted authority. This assumption is consistent with prior work [28], [29], [30] and represents a foundational requirement for protocols in this domain. In contrast to the trust model of C2PA [28], our approach does not impose any additional trust assumptions. Specifically, both the image editor and storage infrastructure are treated as untrusted entities. Importantly, the protocol enables public verification of the integrity proofs without requiring access to the original image. We assume that the adversary $\mathcal{A}$ cannot forge:

- Digital signatures, such as ECDSA or EdDSA on behalf of the tamper-proof camera.
- The underlying hash function used within the system is assumed to be secure, specifically satisfying collision resistance against any probabilistic polynomial-time (PPT) adversary (e.g., Poseidon [20]).
- We assume that it is computationally infeasible to generate a false proof within the employed zero-knowledge proving system (e.g., zkSNARKs), thereby preserving the soundness and integrity of the proof mechanism [31], [32].

4.2.1. Security and Privacy Goals. The design goals of our proposed work are outlined as follows:

1) ***Authenticity***: It ensures that the data being shared, such as traffic alerts or vehicle status is genuine, unaltered, and comes from reliable sources, helping prevent misinformation or malicious tampering.
2) ***Proof of Location***: Proof of Location in SIoV allows a vehicle to prove its presence at a specific location to an RSU, revealing the location but keeping its identity hidden, thereby ensuring location trustworthiness with user anonymity.
3) ***Reliability:*** The protocol ensures data reliability by requiring each witness to submit a signed response, which the responder verifies and aggregates only after reaching a valid threshold. The final result is stored on IPFS and anchored to the blockchain, ensuring integrity, authenticity, and non-repudiation. For location reliability, witnesses collect signatures from nearby RSUs, providing verifiable proof of their physical presence and strengthening the trustworthiness of location claims.
4) ***User Anonymity***: It ensures that the identity of a vehicle or its driver cannot be directly linked to the data they transmit during interactions in the SIoV environment. This protects individuals from tracking, profiling, or surveillance by malicious entities or even honest-but-curious infrastructure.
5) ***Data Privacy***: It ensures that sensitive information such as user identity, facial features, or vehicle number plates remains protected, while still enabling essential verification processes. This approach builds trust without compromising personal or confidential data.
6) ***Fairness:*** The protocol ensures fair participation by requiring all parties to act honestly. A malicious requester cannot access traffic data without providing the agreed incentive, while a dishonest responder cannot earn rewards without submitting valid information. Similarly, witnesses must provide verifiable testimony to receive compensation. This design upholds trust, accountability, and fairness in the data-sharing process.
7) ***Decentralization:*** The protocol functions in a decentralized setting, relying on smart contracts to operate autonomously without the need for a central authority or trusted third party.
8) ***Soundness:*** When all parties behave honestly, the responder and witnesses are rightfully compensated, and the requester gains access to authentic and trustworthy traffic information.

5. Proposed Protocol

In this section, we first describe the proposed scheme, followed by a formal proofs of the statements that a witness vehicle must generate as part of the protocol.

5.1. Protocol Overview

The protocol is illustrated in Figure 1. A requester $\mathcal{R}_{eq}$ (e.g., a vehicle or mobile device) initiates a query to the smart contract to obtain specific information about a particular location. RSUs either periodically poll the smart contract to check for update requests relevant to their coverage areas, or the query may be propagated to RSUs nearest to the target location by a third-party entity. An alternative approach is for RSUs to continuously update the status of their respective locations on-chain, independent of specific requests, allowing vehicles to access the latest data at any time. However, this model lacks incentive mechanisms for participation.

Upon receiving the query (or through proactive status updates), the RSUs broadcast the request to all nearby vehicles within their communication range. If possible, the RSU may also contribute its own observations or sensor data in response to the query, we refer to this RSU as responder $\mathcal{R}_{es}$.

Vehicles present in the vicinity may respond to the query by sharing relevant information. Specifically, each responding vehicle, denoted $\mathcal{W}$, provides an image captured from the street view, which has been redacted to remove privacy-sensitive elements—such as license plates or human faces. Alongside the redacted image, the vehicle generates a ZKP attesting that the redacted image is a valid transformation of an original image with a legitimate digital signature.

To preserve anonymity, $\mathcal{W}$ also includes a ZKP proving that its public key is valid and was previously registered, without revealing the key itself. Since the requester will later verify the results associated with its query, $\mathcal{W}$ must additionally prove that it was physically located within the queried area during the relevant time window. As described later in this section, $\mathcal{W}$ must interact with at least k RSUs (atleast $k \geq 4$)[2] [33], [34] to generate a location proof. This proof is produced through an interactive protocol involving timestamped and signed messages exchanged between the RSUs and the witness vehicle.

Once all proofs are generated, the vehicle transmits the redacted image along with the proofs of image integrity and identity validity back to the RSU that issued the query.

Upon receiving responses from vehicles, each RSU verifies the submitted proofs. After validating the data the verified data is then uploaded to an IPFS server and the corresponding content identifier (CID) is recorded on the blockchain. This enables both the original requester and other authorized third parties to access the verified status of the location through the blockchain. Since storing large data such as redacted images directly on-chain would be expensive, it is more practical to store only the ZK proofs and the associated CID on the blockchain. The actual redacted images are stored off-chain on IPFS or a similar decentralized storage platform, ensuring both cost-efficiency and data availability.

2. In general, secure positioning in one-dimensional space requires at least two verifiers, while positioning in two/three-dimensional space typically necessitates four verifiers to uniquely determine a prover's location. Since the SIoV environment operates in a three-dimensional space, we consider four RSUs in our context. However, our protocol does not enforce a fixed number of verifiers, allowing flexibility based on the specific scenario and application requirements.

5.2. Proving Phase

During this phase, the vehicle is required to prove two statements: (1) the original image α, captured by the camera, is authentic and unaltered, denoted by the proof π_α; and (2) the final submitted image is the result of applying a valid redaction to α using a publicly available mask vector M. In this section, we detail the construction of each proof.

Proof of Authenticity (π_α). To prove the authenticity of the original image α, the vehicle must first demonstrate its physical presence at a specific location. This is accomplished by proving that it successfully interacted with k trusted RSUs and obtained a certificate from each, attesting to its presence within their radar coverage at a given time [35]. Secondly, the vehicle must prove that the original image α is signed with a valid digital signature σ_α. This signature is generated by the vehicle's trusted and tamper-proof camera (or a trusted application like Truepic on the smart phone), which holds the private signing key $sk_{\mathcal{W}}$. However, in order to preserve the vehicle's privacy, the signature σ_α is not revealed directly. Instead, the vehicle proves in zero-knowledge that the signature is valid and corresponds to a public key $pk_{\mathcal{W}}$, which is registered within a verifiable credential vector commitment Δ_{PK} handled by $\mathcal{CA}$. To support this privacy-preserving verification, we adopt a set of anonymity-preserving techniques similar to those proposed in [36].

Definition 3 (π_α)**.** *Let $C_\alpha = H(pk_{\mathcal{W}} \| h_\alpha \| r)$ be the original commitment to the image α that was signed and authenticated by some RSUs. Let σ_α denote the digital signature computed over the image α using the vehicle's private key $sk_{\mathcal{W}}$, which corresponds to the public key $pk_{\mathcal{W}}$. We assume that $pk_{\mathcal{W}}$ is pre-registered in a public vector commitment structure (e.g., a Merkle tree), denoted by Δ_{PK}.*

The statement to be proven is the following:

$$S\big[C_\alpha, h_\alpha, \Delta_{PK}\big] : \Big\{ \exists\, \sigma_\alpha, \delta_{pk}, pk_{\mathcal{W}}, r \;\; s.t.$$
$$C_\alpha = H(pk_{\mathcal{W}} \| h_\alpha \| r)$$
$$\wedge\; \mathcal{V}_\sigma(\sigma_\alpha, h_\alpha, pk_{\mathcal{W}}) = 1 \wedge \mathcal{V}_\Delta(\Delta_{PK}, \delta_{pk}, pk_{\mathcal{W}}) = 1 \Big\}$$

Proof of Redaction (π_β). After establishing the authenticity of the original image α, the next step is to prove that the submitted image β, with hash $h_\beta = H_\phi(\beta)$, is a valid redaction of α using a publicly known mask vector M. The goal is to show that β results from applying a redaction function f_R to α and M, while preserving the integrity of both images through their respective hashes $h_\alpha = H_\phi(\alpha)$ and $h_\beta = H_\phi(\beta)$.

Definition 4 (π_β)**.** *Let α denote the original image captured by the vehicle's camera, and let $h_\alpha = H_\phi(\alpha)$ be its cryp-*

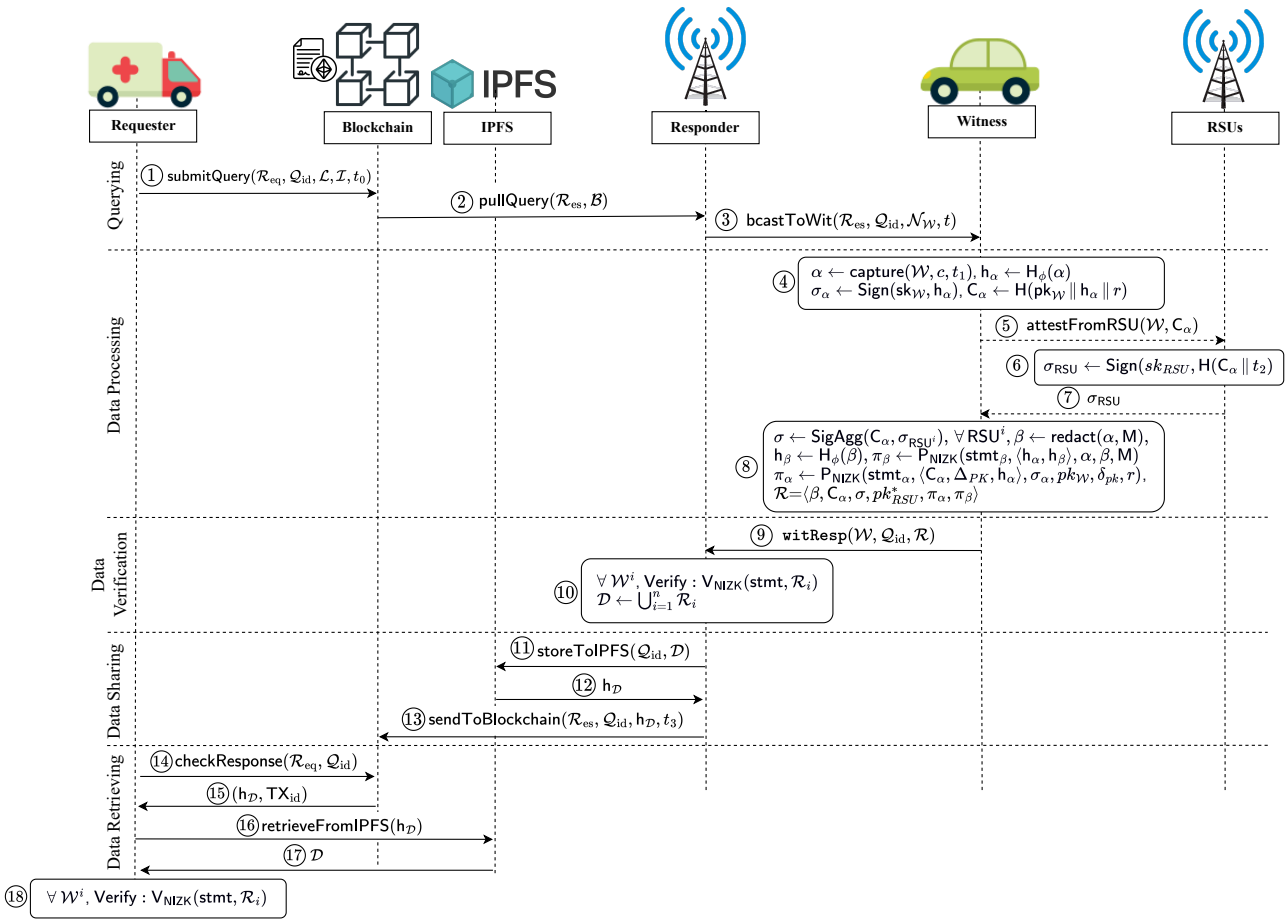

Figure 1. Traffic Management Scenario: The requester $\mathcal{R}_{eq}$ initiates a query for location-specific information. In general scenarios such as road safety alerts or announcement dissemination, the protocol begins from Step 4 onward, skipping Steps 1–3. In such cases, Step 14 is defined by the blockchain lookup function: $\mathsf{queryToBC}(\mathcal{R}_{eq}, \mathcal{Q}_{id}, \mathcal{L}, \mathcal{I}) \rightarrow \{\mathsf{status}, h_\mathcal{D}\}$, which checks for relevant information and returns its availability status along with the corresponding IPFS content identifier.

tographic hash. The vehicle submits a proof along with the redacted image β, satisfying the following statement:

$$S[h_\alpha, h_\beta, M] : \Big\{ \exists\, \alpha, \beta \;\; s.t. \\ h_\alpha = H_\phi(\alpha) \;\wedge\; h_\beta = H_\phi(\beta) \;\wedge\; \beta = f_R(\alpha, M) \Big\}$$

Algorithm 1 defines the logic for realizing the statement of Definition 4 in zero-knowledge. The public inputs to the circuit include the hash of the original image h_α, the hash of the redacted image h_β, and the redaction mask $M \in \{0,1\}^{h \times w}$. The private inputs consist of the block-wise pixel values of the original image, denoted by α. Each image is divided into $h \times w$ blocks, where each block contains 16×16 pixels. Thus, α is represented as an array of $(h \times w)$ blocks of fixed pixel dimensions.

The circuit begins by initializing two hash accumulators, $calc\_h_\alpha$ and $calc\_h_\beta$, to zero. These accumulators are used to recompute the hash values of the original and redacted images within the circuit, ensuring consistency with the publicly provided hash commitments. The main body of the circuit iterates over each block index (i, j) corresponding to the image dimensions. For each block, the hash of the original image is incrementally computed by applying a hash function H_b to each block $\alpha_{i,j}$ and appending the result to the current accumulator. The function H_b is assumed to be a standard array-based hash that operates over an entire 16×16 pixel block.

Before performing the redaction, the circuit enforces that the redaction mask M is properly formatted by asserting the constraint $M_{i,j} \times (M_{i,j} - 1) = 0$ for each block. This ensures that every entry in the mask is binary, i.e., either 0 or 1. The redacted image β is then reconstructed by computing a temporary value $tmp = \alpha_{i,j} \times M_{i,j}$. This enforces that redacted blocks (where $M_{i,j} = 0$) are zeroed out in β, while unredacted blocks (where $M_{i,j} = 1$) preserve their original values from α. The hash accumulator $calc\_h_\beta$ is then updated using the same incremental approach as for $calc\_h_\alpha$. Figure 2 shows the hash data flow of the redaction circuit.

Finally, after processing all blocks, the circuit verifies that the recomputed hash values match the corresponding

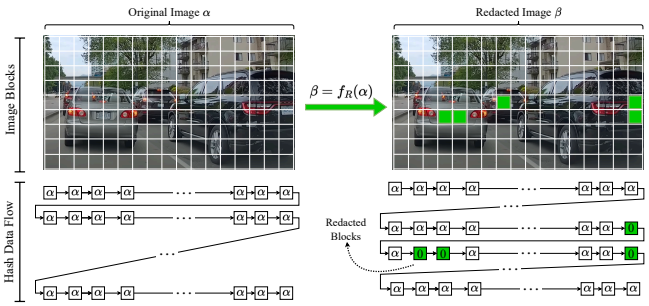

Figure 2. Hash data flow of the redaction. Sample from [37].

public inputs. Specifically, it enforces the constraints $h_\alpha = calc\_h_\alpha$ and $h_\beta = calc\_h_\beta$. These assertions collectively ensure the soundness of the redaction process by confirming that h_β corresponds to the hash of a valid redacted image β, derived from the original image α according to the redaction mask M.

Algorithm 1: Redaction Circuit

Public Input : h_α, h_β, $M_{h \times w}$
Private Input : $\alpha_{h \times w}$
Public Output: h_α, h_β

$calc\_h_\alpha \leftarrow 0$
$calc\_h_\beta \leftarrow 0$
for $i : 0 \rightarrow h$ **do**
  for $j : 0 \rightarrow w$ **do**
    $calc\_h_\alpha \leftarrow H(calc\_h_\alpha | H_b(\alpha_{i,j}))$
    assert $M_{i,j} \times (M_{i,j} - 1) == 0$ // $M_{i,j} \in \{0, 1\}$
    $tmp \leftarrow \alpha_{i,j} \times M_{i,j}$ // redact block based on $M_{i,j}$
    $calc\_h_\beta \leftarrow H(calc\_h_\beta | H_b(tmp))$
assert $h_\alpha == calc\_h_\alpha$
assert $h_\beta == calc\_h_\beta$

5.3. Verification Phase

Upon receiving the proofs from the witness vehicles, the responder $\mathcal{R}_{es}$ initiates the verification phase. To generate the location proof, each witness vehicle first engages with the RSU by sending a hash consisting of its public key, a random nonce, and a timestamp: $H(pk_\mathcal{W} | r | t_i)$. The RSU signs this hash and returns the signed message to the vehicle. Subsequently, the witness vehicle signs the hash of the captured image using the same public key (i.e., the dashcam's public key). It then generates a ZKP demonstrating that the public key ($pk_\mathcal{W}$) used to sign the image hash is identical to the one included in the RSU-signed message at the known timestamp. This establishes that the image was captured at the claimed location and time.

Next, the witness vehicle redacts privacy-sensitive regions of the image (e.g., license plates or faces) and generates a ZKP showing that the redacted image is derived from the original, signed image, with the timestamp included as metadata.

These proofs, along with the redacted image, are submitted to the RSU. The RSU begins verification by checking the validity of the image hash's signature. It then verifies that the public key used for signing belongs to a valid registry (e.g., via Merkle proof with root Δ_{PK}). If the public key is deemed valid, the RSU proceeds to verify the correctness of the redaction proof with respect to the signed original image hash value.

5.4. Soundness of the Protocol

The soundness of the proposed protocol follows directly from the security properties of its underlying cryptographic components. In particular, the protocol's correctness reduces to the collision resistance of the employed hash functions, the soundness of the digital signature schemes, and the unforgeability of the zkSNARK constructions—both the general-purpose and folding-based variants. Each of these primitives is assumed to satisfy standard security definitions, and under these assumptions, the protocol does not allow a malicious prover to produce a valid proof for a false statement.

Theorem 1. *The probability that a probabilistic PPT adversary successfully breaks the soundness of the statement defined in Definition 3 is negligible.*

Proof sketch. To break the soundness of the statement in Definition 3, a probabilistic polynomial-time (PPT) adversary $\mathcal{P}^*$ must succeed in one of the following scenarios: (1) The adversary produces a forged proof π^*_α for a false statement, meaning that one or more constraints defined by the statement are violated. For instance, the computation of the commitment $C_\alpha = H(pk_\mathcal{W} \| h_\alpha \| r)$ may be incorrect, the signature σ_α may fail verification, or the public key $pk_\mathcal{W}$ may not be included in the set Δ_{PK}. Nonetheless, the adversary outputs a proof that passes verification. This contradicts the assumed soundness of the underlying zkSNARK construction. (2) The adversary is able to open the commitment C_α to values $(pk'_\mathcal{W}, h'_\alpha, r')$ that differ from the original ones. Achieving this would require breaking the binding property of the commitment scheme. In our setting, this would imply finding a collision in the hash function H, violating its assumed collision resistance. (3) The adversary succeeds in passing verification checks of the components $\mathcal{V}_\sigma$ and $\mathcal{V}_\Delta$, both of which are assumed to be sound against PPT adversaries.

Thus, we argue the overall probability that $\mathcal{P}^*$ succeeds is negligible in the security parameter λ.

$$\Pr\left[\begin{array}{l} \Big(\mathcal{V}_\Delta(\Delta_{PK},\delta_{pk},pk_{\mathcal{W}})\neq 1 \\ \vee\big(\mathcal{V}_\Delta(\Delta_{PK},\delta_{pk},pk_{\mathcal{W}})=1 \\ \wedge\mathcal{V}_\Delta(\Delta_{PK},\delta_{pk},pk'_{\mathcal{W}})=1 \\ \wedge pk_{\mathcal{W}}\neq pk'_{\mathcal{W}}\big) \\ \vee\big(C_\alpha=H(pk'_{\mathcal{W}}\|h'_\alpha\|r') \\ \wedge(pk'_{\mathcal{W}}\neq pk_{\mathcal{W}}\vee h'_\alpha\neq h_\alpha)\big)\vee \\ (\mathcal{V}_\sigma(\sigma'_\alpha,h'_\alpha,pk'_{\mathcal{W}})=1\wedge h'_\alpha\neq h_\alpha)\Big) \\ \wedge\,\mathcal{V}_\pi(vk,\langle C_\alpha,h_\alpha,\Delta_{PK}\rangle,\pi_\alpha)=1 \end{array}\middle|\begin{array}{c} pp\leftarrow\mathcal{G}(1^\lambda), \\ (pk,vk)\leftarrow\mathcal{K}(pp,S), \\ (\delta_{pk},\sigma_\alpha,\sigma'_\alpha,pk'_{\mathcal{W}},h'_\alpha, \\ r',pk_{\mathcal{W}},h_\alpha,r)\leftarrow\mathcal{P}^* \\ C_\alpha\leftarrow H(pk_{\mathcal{W}}\|h_\alpha\|r) \\ \pi_\alpha\leftarrow\mathcal{E}(pp,\rho) \end{array}\right]=\text{negl}(1^\lambda)$$

□

Theorem 2. *The probability that a probabilistic polynomial-time (PPT) adversary successfully breaks the soundness of the statement defined in Definition 4 is negligible.*

Proof sketch. Let β' denote the redacted image produced by the PPT adversary $\mathcal{P}^*$, and let β be the correct redacted image derived from the original image α. Let H_R represent the folding function applied during the redaction process, corresponding to lines 4 through 8 of Algorithm 1. The soundness argument proceeds in two parts. (1) First, suppose the adversary submits a proof that verifies under valid public parameters, but the revealed redacted image β' differs from the true β, or the claimed original image α' differs from α, while still satisfying $H_\phi(\beta')=H_\phi(\beta)=h^n_\beta$ or $H_\phi(\alpha')=H_\phi(\alpha)=h^n_\alpha$. In this case, the adversary must have found a collision in the hash function H_ϕ, and by extension in H, which contradicts the assumption that H is collision-resistant. (2) Second, consider the case where the adversary succeeds in verifying a malformed proof Π' using public parameters other than the expected values $z=\{h^n_\alpha,h^n_\beta\}$ and $z_0=\{0,0\}$. Such an event would imply a violation of the soundness property of the underlying folding scheme. However, the folding-based zkSNARK construction guarantees that the probability of a PPT adversary achieving this is negligible.

Therefore, the overall probability of a successful soundness violation is negligible in the security parameter λ.

$$\Pr\left[\begin{array}{l} \Big(\big(\alpha'\neq\alpha\wedge H_\phi(\alpha')=h_\alpha\big) \\ \vee\big(\beta'\neq\beta\wedge H_\phi(\beta')=h_\beta\big) \\ \vee\big(z_0=\{0,0\} \\ \wedge\, z\neq\{h_\alpha,h_\beta\}\big)\Big) \\ \wedge\,\mathcal{V}(vk,n,z_0,z,\Pi)=1 \end{array}\middle|\begin{array}{c} pp\leftarrow\mathcal{G}(1^\lambda) \\ \alpha',\beta',M,(z_0,z,\Pi)\leftarrow\mathcal{P}^*(pp;\rho) \\ (pk,vk)\leftarrow\mathcal{K}(pp,H_R) \\ (\omega_0,\ldots,\omega_{n-1})\leftarrow\mathcal{E}(pp,z_0,z;\rho) \\ \omega_i=\{\alpha_i,M_i\}\ \forall i\in\mathbb{N}_n \\ z_i\leftarrow H_R(z_{i-1},\omega_{i-1})\ \forall i\in\mathbb{N}_n \end{array}\right]=\text{negl}(1^\lambda)$$

□

6. Implementation

We begin this section by outlining our implementation of a proof-of-concept prototype, which is available in an open-source GitHub repository[3]. We then evaluate the overall performance of the proposed framework, focusing on both the computational and communication complexity incurred by the involved parties.

3. https://github.com/----anonymized-for-review-----

6.1. Implementation Details and System Setup

6.1.1. Proof π_α. To construct the proof π_α, we employ a general zkSNARK scheme, specifically Groth16 [38]. The constraints corresponding to the statement defined in Definition 3 are implemented using the Circom [39]. Within this framework, the verification component $\mathcal{V}_\sigma$ is realized using a Schnorr signature verification circuit. Additionally, the component $\mathcal{V}_\Delta$ is implemented using standard Merkle tree inclusion circuits based on Poseidon hash [20] to validate membership proofs efficiently.

6.1.2. Proof π_β. At the core of our construction, we require a trustless computation of the chained hashes for both the original and redacted images, along with a ZKP attesting to the correctness of this computation. Recent work [36] has demonstrated that folding-based zkSNARKs, such as Nova [21], [40], are particularly well-suited for this task, especially under constrained hardware settings. These schemes are most effective when the computation can be structured as repeated applications of the same function, as illustrated in Figure 2. As discussed Section 5, we employ folding-based zkSNARKs to generate proofs that attest a redacted image is derived from a specific original image. Similar approaches have been explored in recent work [36], [41]. However, a key distinction in our approach is the use of compressed JPEG data, in contrast to prior work that operates on raw pixel-level data.

Our method focuses on the redaction of specific blocks, which allows us to process each JPEG block independently, treating it as a standalone JPEG image. For example, assuming a redaction granularity of 8×8 blocks, an 800×800 pixel image can be treated as 10,000 independent JPEG images. Redaction is then applied at this block level. However This design enables our zkSNARK circuits to operate directly on compressed JPEG data represented as arrays, which results in significantly lower circuit complexity compared to previous work [36], [41].

Our implementation for this proof, leverages the `Nova-snark` library [40], which provides an optimized realization of the Nova proof system and supports the generation of a compact final zkSNARK proof using Spartan [42]. To define the underlying arithmetic constraints, we utilize the Circom domain-specific language [39]. These Circom circuits are then compiled into a form compatible with `Nova-snark` using the `nova-scotia` interface [43].

6.1.3. System Setup. We conducted our evaluations on a commodity device: a laptop equipped with Ryzen 9 6800 CPU and 16 GB of RAM, running a Linux environment via Windows Subsystem for Linux (WSL).

6.2. Performance Analysis

6.2.1. Proof π_α. Table 2 summarizes the key performance metrics of our prototype for this phase of the protocol. Notably, the proving time remains relatively low regardless of the size of the anonymity set in which $pk_{\mathcal{W}}$ resides. This

TABLE 2. PERFORMANCE ANALYSIS OF PROOF π_α

Number of Public keys	Private Inputs	Public Inputs	Prove Time	Verifier Time	Proof Size
2^{10}	13	3	∼1 s	< 0.2 s	806 B
2^{20}	23	3	1.5 s	< 0.2 s	806 B

indicates that the number of public keys included in the Merkle tree does not significantly affect prover performance. Moreover, both the verification time and proof size remain constant due to the succinct verification characteristics of the Groth16 proving system.

6.2.2. Proof π_β. The Figure 6.2.2 reports the prove times for redactions using 32×32 versus 64×64 block sizes across various JPEG compression levels, resolutions, and color modes. As expected, Full HD (FHD) images incur higher proving costs compared to HD images due to the larger number of blocks involved. For example, the prove time for redacting a color FHD image with 50% JPEG compression reaches 182 seconds, whereas its HD counterpart requires only 82 seconds. Results show that JPEG compression level has a modest effect on performance, with some minor variations observed. The relative difference between color and grayscale images varies by configuration but overall, it is minimal. Finally, while image characteristics influence performance, the granularity of the redaction remains the key computational factor.

Across all configurations—both HD and FHD, color and grayscale—the proving times of 64×64 block sizes are significantly lower than those in the 32×32 case, falling below 60 seconds in all scenarios. HD images require only 23 seconds to redact, and FHD images stay around 50 seconds, with negligible differences between color modes or compression levels. This reduction is attributed to the coarser redaction granularity: larger blocks reduce the number of independent proof segments and simplify circuit logic. However, this efficiency gain comes at the cost of reduced redaction precision, since larger blocks encompass more image content and limit fine-grained control over which regions can be hidden.

Across all evaluated cases, the verification time remained within the range of 50 to 70 milliseconds, and the compressed proof size was consistently approximately 13 KB.

6.2.3. Storage and Communication Complexity. Table 3 presents a detailed breakdown of the storage requirements for each public value and two proofs (π_α and π_β). In total, the combined storage and communication overhead for these elements is approximately 14 KB. When compared to typical original image sizes (JPEG images with HD resolution) ranging from 0.5 MB to 2 MB, this overhead represents roughly 1.5% of the original data size, indicating minimal impact on overall data transmission and storage efficiency.

7. Related Work

Here, we categorize the related work into two groups: those focused primarily on the reliability and authenticity of

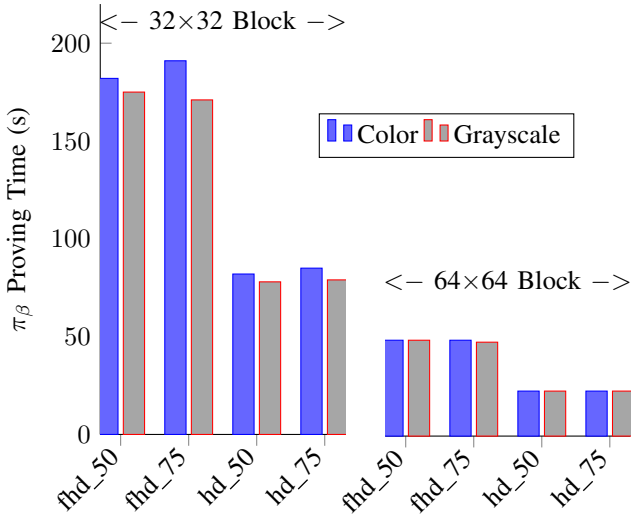

Figure 3. Comparison of folding-based proof generation times for π_β by resolution, JPEG compression, and block size.

TABLE 3. COMMUNICATION AND STORAGE COMPLEXITY BREAKDOWN

Element	π_α	π_β	h_α	h_β
Size	806 B	13 KB	32 B	32 B

Element	σ_C	pk^* e.g. 4 RSUs	Δ_{PK}	C_α
Size	64 B	32 B - 128 B	32 B	32 B

Total	Image	Overhead
∼14 KB	0.5 MB-2 MB	∼1.5%

shared information, and those emphasizing privacy aspects such as location confidentiality and user anonymity. Table 4 provides a comparative overview of the security and privacy features of the proposed TrafficProof scheme against existing data sharing protocols.

7.1. Reliable Information Sharing

Reliable data sharing in the SIoV is critical for enabling secure, privacy-preserving, and efficient communication among vehicles and infrastructure. The Proof of Traffic Flow Condition (PoTC) consensus mechanism [46] aims to provide secure consensus in Internet of Vehicles (IoV) using traffic flow data to provide reliability scores for participants. Based on their results, PoTC can tolerate a larger proportion of malicious RSUs (MRs) compared to Practical Byzantine Fault Tolerance (PBFT) consensus, and also secure against invalid voting and traffic-flow data tampering. Limitations include increased computational difficulty in training phase, computational cost of node assessment, and latency, additional communication complexity at both the RSU and LRSU layers. The SEDS scheme [14] provides several security features including confidentiality of vehicle sensory data through Paillier homomorphic encryption and a shared key between the vehicle and RSU. It also offers location privacy preservation using an autonomous and controllable dynamic pseudonym mechanism. By employing blockchain

TABLE 4. COMPARISON AMONG THE STATE-OF-THE-ART WORKS

Protocol	Data Reliability	User Anonymity	Data Privacy	Location Privacy	Proof of Location	Public Verifiability	Data Authenticity
Xia et al. [18]	✗	✓*	✗	✗	✗	✗	✗
Wang et al. [19]	✓	✓*	✗	✗	✗	✓	✗
Xia et al. [44]	✗	✗	✗	✓	✗	✗	✗
Min et al. [13]	✗	✗	✗	✓	✗	✗	✗
Xu et al. [45]	✗	✗	✗	✓	✗	✗	✗
Chen et al. [15]	✗	✓*	✗	✓	✗	✓	✗
Li et al. [9]	✓	✓*	✗	✗	✗	✓	✗
Mohanty et al. [11]	✓	✓	✗	✗	✗	✗	✗
Hu et al. [14]	✓	✓	✓	✓	✗	✓	✗
TrafficProof (Ours)	✓	✓	✓	N/A	✓	✓	✓

* Conditional Anonymity (with Traceability)

technology to store data analysis results, SEDS ensures that the shared data remains tamper-proof and resistant to attacks by malicious entities, thus guaranteeing data integrity and preventing unauthorized modifications. Although there is still a computational overhead on vehicles for encrypting sensory data. Authors in [18] provide conditional privacy protection and secure data sharing. The scheme utilizes ciphertext-policy attribute-based encryption (CP-ABE) to achieve anonymous one-to-many sharing of data with fine-grained access management. The trusted authority (TA) has the ability to collaboratively trace the actual identities of malicious vehicles according to the data stored in the blockchain system. A lightweight ledger-based blockchain system is designed to record data indexes, data ciphertexts, and shared records, aiming to reduce the storage and communication overhead of nodes in data sharin. The proposed scheme incur more overhead in generating the message because it uses symmetric encryption to encrypt the message and attribute-based encryption to encrypt the symmetric key. Although a lightweight ledger is proposed, the paper mentions that for optimal efficiency, it is advisable to limit the number of nodes in the blockchain to 20 or fewer. The security of the scheme relies on the assumption that the trusted authority (TA) can be fully trusted.

Authors in [19] address the challenges of secure and efficient information sharing in vehicular networks. they introduce a group signature scheme tailored for reliable anonymous announcements. Building upon this, they present a new threshold anonymous announcement protocol that disseminates compressed announcements with batched endorsements. challenges in scenarios such as network unreliability where vehicles might disconnect after sending initial commitments and to handle unreliable witnesses who might send bogus endorsements.

7.2. Location-Specific Information Sharing

Location-specific information sharing in the SIoV poses unique challenges in balancing data utility with strong privacy guarantees. Several recent works have addressed these challenges by introducing innovative privacy-preserving mechanisms tailored to vehicular environments. Xia et al. [44] introduce a reinforcement learning-based approach to privacy preservation in spatial crowdsourcing for VANETs. Their scheme dynamically adjusts each vehicle's privacy level based on environmental context and individual user needs. Laplacian noise is applied during task allocation to protect location privacy while maintaining high allocation accuracy. However, this dynamic personalization introduces complexity in system management, imposes computational overhead on vehicles, and relies on real-time feedback and historical data for effective policy adaptation. In a related line of work, Xu et al. [45] propose a scheme leveraging Personalized Differential Privacy (PDP) to tailor location protection based on user-specific sensitivity. Privacy levels are adjusted according to a "sensitivity distance," enabling a balance between data utility and privacy protection. The scheme formulates a multi-objective optimization model to allocate privacy budgets, enhancing Quality of Service (QoS). Despite its strengths, it depends heavily on accurate sensitivity assessments and remains susceptible to adversarial attacks, such as Bayesian inference, depending on noise levels and attacker knowledge. Min et al. [13] present SAGEO, a semantic-aware privacy-preserving mechanism that adapts privacy protection based on the sensitivity of various location types (e.g., hospitals vs. parks). Using curved-space modeling, it introduces greater noise in sensitive areas to enhance privacy while minimizing it in less sensitive regions to preserve QoS. Unlike traditional approaches, SAGEO achieves differential privacy without a trusted third party. However, it primarily targets scenarios involving a single sensitive location and incurs higher execution time due to the added complexity of semantic sensitivity modeling. Chen et al. [15] propose SAVE, a dual-level location privacy scheme. Level I protection is achieved using a one-way trapdoor function, while Level II relies on zero-knowledge range proofs built on lightweight one-way hash chains. The scheme is resilient to dynamic physical tracing attacks, where adversaries attempt to infer vehicle identities by tracking their movement across locations. Through online/offline aggregate signatures and LBS bundle filtering, SAVE reduces verification overhead and suits resource-limited onboard units (OBUs). Still, practical deployment may be hindered by system complexity and re-

liance on subjective user-defined privacy parameters, which could challenge real-world applicability.

Lastly, Cai et al. [16] investigate user privacy perceptions in Connected Autonomous Vehicles (CAVs) using Vehicle-to-Everything (V2X) communication. Through a large-scale user study involving 595 participants across multiple V2X application scenarios, they analyze decision-making behaviors and privacy concerns. Their findings underscore the importance of educating users about the privacy implications of data sharing and recommend transparent communication from service providers to foster informed user decisions.

Our approach to achieve proof-of-location fundamentally differs from conventional location privacy schemes. While existing methods typically aim to hide the location of vehicles to protect their identity, this can reduce the reliability and verifiability of the shared data. In our approach, the witness vehicle's location is made public, which improves transparency and helps ensure data authenticity and trustworthiness. Importantly, we preserve privacy not by hiding the location, but by fully anonymizing the witness's identity. This decoupling of identity and location enables a trustless and verifiable data sharing process. Moreover, our design allows witnesses to redact information in a trustless manner, further preserving privacy without requiring a trusted third party. As a result, our framework offers highly accurate and reliable traffic data, while fully preserving the privacy of witness.

8. Discussion and Future Work

This paper introduces a novel protocol for reliable traffic information sharing that fully preserves the anonymity and privacy of participating witnesses. The protocol enables witness vehicles to generate ZKPs attesting that the shared data is the result of a valid redaction process applied to an authentic original source. Additionally, the authenticity of the original data and the identity of the witness are concealed using a separate ZKP, ensuring that both remain private. As a result, the protocol supports traffic data sharing that is not only anonymous and privacy-preserving but also publicly verifiable, reliable, and accurate—benefiting from the transparency and integrity of location-bounded inputs.

While the proposed framework addresses several key challenges, there remain open directions for future research. First, the current system relies on trusted hardware (e.g., tamper-proof cameras) to ensure original image authenticity. A promising avenue is to eliminate this dependency by developing purely cryptographic methods that offer equivalent guarantees without specialized hardware. Second, the current solution employs an interactive protocol for location verification. A valuable extension would be the design of a non-interactive proof-of-location mechanism, which could improve scalability and reduce communication overhead while maintaining strong security guarantees. Third, the system currently supports still-image capture for event reporting. Enhancing it to handle 360-degree video streams would enable richer contextual evidence and continuous scene understanding. Fourth, the current incentive mechanism is relatively simple and static; future work could explore more adaptive and robust models that reward witnesses based on the quality, timeliness, and consistency of their contributions. Lastly, while the framework is primarily focused on traffic information sharing, its core mechanisms can be extended to a wide range of SIoV scenarios, such as emergency incident response, pollution and hazard detection, or secure vehicular services.

Acknowledgments

The authors used ChatGPT4 only for grammatical revision of the text in the paper to correct any typos, grammatical errors, and awkward phrasing. This work was supported by the European Research Council (ERC) under the European Union's Horizon 2020 innovation program (grant PROCONTRA-885666).

References

[1] X. Wang, Z. Ning, M. Zhou, X. Hu, L. Wang, Y. Zhang, F. R. Yu, and B. Hu, "Privacy-preserving content dissemination for vehicular social networks: Challenges and solutions," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 2, pp. 1314–1345, 2019.

[2] M. B. Mollah, J. Zhao, D. Niyato, Y. L. Guan, C. Yuen, S. Sun, K.-Y. Lam, and L. H. Koh, "Blockchain for the internet of vehicles towards intelligent transportation systems: A survey," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4157–4185, 2021.

[3] I. Ullah, M. Ali Shah, A. Khan, and M. Guizani, "Location privacy schemes in vehicular networks: Taxonomy, comparative analysis, design challenges, and future opportunities," *ACM Computing Surveys*, vol. 57, no. 6, pp. 1–44, 2025.

[4] L. Benarous, S. Zeadally, S. Boudjit, and A. Mellouk, "A review of pseudonym change strategies for location privacy preservation schemes in vehicular networks," *ACM Computing Surveys*, vol. 57, no. 8, pp. 1–37, 2025.

[5] S. Banerjee, D. Das, P. Chatterjee, B. Blakely, and U. Ghosh, "A blockchain-enabled sustainable safety management framework for connected vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 6, pp. 5271–5281, 2023.

[6] X. Wang, Z. Ning, X. Hu, L. Wang, B. Hu, J. Cheng, and V. C. Leung, "Optimizing content dissemination for real-time traffic management in large-scale internet of vehicle systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1093–1105, 2018.

[7] K. C. Dey, A. Mishra, and M. Chowdhury, "Potential of intelligent transportation systems in mitigating adverse weather impacts on road mobility: A review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1107–1119, 2014.

[8] J. Ni, K. Zhang, Y. Yu, X. Lin, and X. Shen, "Privacy-preserving smart parking navigation supporting efficient driving guidance retrieval," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 6504–6517, 2018.

[9] L. Li, J. Liu, L. Cheng, S. Qiu, W. Wang, X. Zhang, and Z. Zhang, "Creditcoin: A privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 7, pp. 2204–2220, 2018.

[10] Y. Zhang, J. Weng, J. Weng, M. Li, and W. Luo, "Onionchain: Towards balancing privacy and traceability of blockchain-based applications," *arXiv preprint arXiv:1909.03367*, 2019.

[11] S. K. Mohanty and S. Tripathy, "Siovchain: time-lock contract based privacy-preserving data sharing in siov," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 24 071–24 082, 2022.

[12] B. S. Naik, S. Tripathy, and S. K. Mohanty, "Musigrdt: Multisig contract based reliable data transmission in social internet of vehicle," in *GLOBECOM 2022-2022 IEEE Global Communications Conference*. IEEE, 2022, pp. 1763–1768.

[13] M. Min, H. Zhu, S. Li, H. Zhang, L. Xiao, M. Pan, and Z. Han, "Semantic adaptive geo-indistinguishability for location privacy protection in mobile networks," *IEEE Transactions on Vehicular Technology*, 2024.

[14] P. Hu, X. Chu, K. Zuo, T. Ni, D. Xie, Z. Shen, F. Chen, and Y. Luo, "Security-enhanced data sharing scheme with location privacy preservation for internet of vehicles," *IEEE Transactions on Vehicular Technology*, 2024.

[15] Y. Chen, T. Zhou, J. Zhou, Z. Cao, X. Dong, and K.-K. R. Choo, "Save: Efficient privacy-preserving location-based service bundle authentication in self-organizing vehicular social networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 11 752–11 766, 2021.

[16] Z. Cai and A. Xiong, "Understand users' privacy perception and decision of {V2X} communication in connected autonomous vehicles," in *32nd USENIX Security Symposium (USENIX Security 23)*, 2023, pp. 2975–2992.

[17] X. Liu, J. Cui, J. Zhang, R. Yin, H. Zhong, L. Wei, I. Bolodurina, and D. He, "Bast: Blockchain-assisted secure and traceable data sharing scheme for vehicular networks," *IEEE Transactions on Information Forensics and Security*, 2025.

[18] Z. Xia, J. Man, K. Gu, X. Li, and L. Huang, "Conditional data-sharing privacy-preserving scheme in blockchain-based social internet of vehicles," *IEEE Transactions on Sustainable Computing*, 2024.

[19] Y. Wang, L. Y. Zhang, X. Zhang, X. Wang, F. Wang, S. Hu, and R. Doss, "Towards threshold anonymous announcement with batch verification for cooperative intelligent transport systems," *IEEE Transactions on Vehicular Technology*, 2024.

[20] L. Grassi, D. Khovratovich, C. Rechberger, A. Roy, and M. Schofnegger, "Poseidon: A new hash function for zero-knowledge proof systems." in *USENIX Security Symposium*, vol. 2021, 2021.

[21] A. Kothapalli, S. Setty, and I. Tzialla, "Nova: Recursive zero-knowledge arguments from folding schemes," in *Annual International Cryptology Conference*. Springer, 2022, pp. 359–388.

[22] E. N. Tas and D. Boneh, "Vector commitments with efficient updates," *arXiv preprint arXiv:2307.04085*, 2023.

[23] D. Catalano and D. Fiore, "Vector commitments and their applications," in *Public-Key Cryptography–PKC 2013: 16th International Conference on Practice and Theory in Public-Key Cryptography, Nara, Japan, February 26–March 1, 2013. Proceedings 16*. Springer, 2013, pp. 55–72.

[24] S. Ebrahimi and P. Hassanizadeh, "From interaction to independence: zksnarks for transparent and non-interactive remote attestation," *The Network and Distributed System Security (NDSS) Symposium*, 2024.

[25] W. Arthur, D. Challener, and K. Goldman, *A practical guide to TPM 2.0: Using the new trusted platform module in the new age of security*. Springer Nature, 2015.

[26] "Sony unlocks in-camera forgery-detection technology," https://www.sony.eu/presscentre/sony-unlocks-in-camera-forgery-proof-technology, 2022, accessed: 2024-01-08.

[27] "TruePic: Secure content transparency with C2PA," https://truepic.com/, 2024, accessed: 2024-01-03.

[28] "C2PA Technical Specification," https://c2pa.org/specifications/specifications/1.2/index.html, 2024, accessed: 2024-01-03.

[29] D. Kang, T. Hashimoto, I. Stoica, and Y. Sun, "ZK-IMG: Attested images via zero-knowledge proofs to fight disinformation," *arXiv preprint arXiv:2211.04775*, 2022.

[30] A. Naveh and E. Tromer, "Photoproof: Cryptographic image authentication for any set of permissible transformations," in *2016 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2016, pp. 255–271.

[31] M. Blum, P. Feldman, and S. Micali, "Non-interactive zero-knowledge and its applications," in *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*, 2019, pp. 329–349.

[32] J. Groth, "On the size of pairing-based non-interactive arguments," in *Advances in Cryptology–EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II 35*. Springer, 2016, pp. 305–326.

[33] N. Chandran, V. Goyal, R. Moriarty, and R. Ostrovsky, "Position based cryptography," in *Annual International Cryptology Conference*. Springer, 2009, pp. 391–407.

[34] ——, "Position-based cryptography," *SIAM Journal on Computing*, vol. 43, no. 4, pp. 1291–1341, 2014.

[35] F. Boeira, M. Asplund, and M. P. Barcellos, "Vouch: A secure proof-of-location scheme for vanets," in *Proceedings of the 21st ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2018, pp. 241–248.

[36] S. Dziembowski, S. Ebrahimi, and P. Hassanizadeh, "Vimz: Private proofs of image manipulation using folding-based zksnarks," *Proceedings on Privacy Enhancing Technologies*, 2025.

[37] "Viofo a119 v3 sample," https://youtu.be/m5BFMn56sos, 2024, accessed: 2025-05-02.

[38] J. Groth, "On the size of pairing-based non-interactive arguments," in *Advances in Cryptology – EUROCRYPT 2016*, M. Fischlin and J.-S. Coron, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 305–326.

[39] M. Bellés-Muñoz, M. Isabel, J. L. Muñoz-Tapia, A. Rubio, and J. Baylina, "Circom: A circuit description language for building zero-knowledge applications," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–18, 2022.

[40] "Nova high-speed recursive arguments from folding schemes," https://github.com/microsoft/Nova, 2024, accessed: 2024-01-03.

[41] P. Mikołajczyk, P. Hassanizadeh, and S. Ebrahimi, "Towards trustless provenance: A privacy-preserving framework for on-chain media verification," Cryptology ePrint Archive, Paper 2025/1024, 2025. [Online]. Available: https://eprint.iacr.org/2025/1024

[42] S. Setty, "Spartan: Efficient and general-purpose zkSNARKs without trusted setup," Cryptology ePrint Archive, Paper 2019/550, 2019, https://eprint.iacr.org/2019/550. [Online]. Available: https://eprint.iacr.org/2019/550

[43] "nova-scotia," https://github.com/nalinbhardwaj/Nova-Scotia, 2024, accessed: 2024-01-03.

[44] Y. Xia, H. Wang, T. Cao, X. Liu, and Z. Liu, "Personalized privacy preserving for spatial crowdsourcing by reinforcement learning in vanets," *IEEE Internet of Things Journal*, 2024.

[45] C. Xu, L. Luo, Y. Ding, G. Zhao, and S. Yu, "Personalized location privacy protection for location-based services in vehicular networks," *IEEE Wireless Communications Letters*, vol. 9, no. 10, pp. 1633–1637, 2020.

[46] Y. Zhao, N. Ding, Y. Hao, and L. Xu, "Potc: A proof of traffic-flow condition consensus for secure and efficient blockchain in the internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

Appendix

A. Functional Specification of TrafficProof

$$\texttt{submitQuery}(\mathcal{R}_{\text{eq}}, \mathcal{Q}_{\text{id}}, \mathcal{L}, \mathcal{I}, t_0)$$

Purpose: This function allows a requester $\mathcal{R}_{\text{eq}}$ to initiate a location- and time-specific query for contextual information in the Social Internet of Vehicles (SIoV) environment.

Input Parameters:

- $\mathcal{R}_{\text{eq}}$: Identity of the requester vehicle initiating the query.
- $\mathcal{Q}_{\text{id}}$: A unique query identifier specifying the type of information requested (e.g., accident report, traffic congestion).
- $\mathcal{L}$: Target location or area of interest for the query.
- $\mathcal{I}$: Incentive value promised by the requester to responders or witnesses.
- t_0: Timestamp marking the moment of query submission.

$$\texttt{pullQuery}(\mathcal{R}_{\text{es}}, \mathcal{B})$$

Purpose: This function enables a potential responder $\mathcal{R}_{\text{es}}$ to retrieve active query messages from the blockchain or a designated query pool $\mathcal{B}$ in order to evaluate whether to participate in responding.

Input Parameters:

- $\mathcal{R}_{\text{es}}$: Identity of the responder vehicle that intends to scan or respond to existing queries.
- $\mathcal{B}$: The blockchain or query repository where all recent or valid queries are stored.

$$\texttt{bcastToWit}(\mathcal{R}_{\text{es}}, \mathcal{Q}_{\text{id}}, \mathcal{N}_{\mathcal{W}}, t)$$

Purpose: This function allows the responder $\mathcal{R}_{\text{es}}$ to broadcast a selected query $\mathcal{Q}_{\text{id}}$ to a nearby set of witnesses $\mathcal{N}_{\mathcal{W}}$, requesting their participation and support in the data verification process.

Input Parameters:

- $\mathcal{R}_{\text{es}}$: Identity of the responder vehicle that initiates the witness broadcast.
- $\mathcal{Q}_{\text{id}}$: The unique identifier of the query being processed.
- $\mathcal{N}_{\mathcal{W}}$: The set of neighboring witness vehicles identified for potential collaboration.
- t: The minimum threshold number of witnesses required to generate valid aggregated support.

$$\texttt{attestFromRSU}(\mathcal{W}, \mathsf{C}_{\alpha})$$

Purpose: This function enables a witness $\mathcal{W}$ to obtain location attestation from a nearby Roadside Unit (RSU), ensuring that its claimed position at that time is verified through a signed credential C_{α}.

Input Parameters:

- $\mathcal{W}$: Identity of the witness vehicle seeking location verification.
- C_{α}: A context-dependent commitment (e.g., based on time/location) for which the RSU is expected to issue a signature.

$$\texttt{witResp}(\mathcal{W}, \mathcal{Q}_{\text{id}}, \mathcal{R})$$

Purpose: This function defines how a witness $\mathcal{W}$ responds to a received query $\mathcal{Q}_{\text{id}}$ by generating and sending a signed response $\mathcal{R}$ back to the responder.

Input Parameters:

- $\mathcal{W}$: The identity of the witness vehicle providing the response.
- $\mathcal{Q}_{\text{id}}$: The identifier of the query received from the responder.
- $\mathcal{R}$: The witness's response message, which includes proof of presence, timestamp, and other relevant information.

$$\texttt{storeToIPFS}(\mathcal{Q}_{\text{id}}, \mathcal{D})$$

Purpose: This function allows the responder to store aggregated data $\mathcal{D}$, including witness responses and supporting evidence on the InterPlanetary File System (IPFS), and associate it with a specific query $\mathcal{Q}_{\text{id}}$.

Input Parameters:

- $\mathcal{Q}_{\text{id}}$: The unique identifier of the original query to which the stored data corresponds.
- $\mathcal{D}$: The complete data bundle to be stored, including verified witness responses, timestamps, signatures, and any contextual metadata.

$$\texttt{sendToBlockchain}(\mathcal{R}_{\text{es}}, \mathcal{Q}_{\text{id}}, \mathsf{h}_{\mathcal{D}}, t_3)$$

Purpose: This function enables the responder $\mathcal{R}_{\text{es}}$ to anchor the reference to verified and aggregated data identified by IPFS hash $\mathsf{h}_{\mathcal{D}}$ on the blockchain, binding it to the original query $\mathcal{Q}_{\text{id}}$ at time t_2.

Input Parameters:

- $\mathcal{R}_{\text{es}}$: Identity of the responder who submits the data.
- $\mathcal{Q}_{\text{id}}$: Identifier of the query to which the data corresponds.
- $\mathsf{h}_{\mathcal{D}}$: IPFS content identifier (CID) or hash representing the data bundle stored off-chain.
- t_3: Timestamp marking the moment of submission to the blockchain.

$$\texttt{checkResponse}(\mathcal{R}_{\text{eq}}, \mathcal{Q}_{\text{id}})$$

Purpose: This function allows the requester $\mathcal{R}_{\text{eq}}$ to check whether a valid response has been submitted to the blockchain for a previously issued query $\mathcal{Q}_{\text{id}}$.

Input Parameters:

- $\mathcal{R}_{\text{eq}}$: Identity of the requester who originally submitted the query.

- $\mathcal{Q}_{\text{id}}$: Identifier of the query for which a response is being checked.

$$\texttt{retrieveFromIPFS}(\mathsf{h}_{\mathcal{D}})$$

Purpose: This function allows any participant (e.g., the requester or verifier) to retrieve the stored data from the InterPlanetary File System (IPFS) using the content identifier (CID) $\mathsf{h}_{\mathcal{D}}$ obtained from the blockchain.

Input Parameter:

- $\mathsf{h}_{\mathcal{D}}$: The IPFS content identifier (CID) referencing the off-chain data bundle associated with a specific query response.