

CreV: A Lightweight Challenge–Response Exchange Protocol and Frame Design for Secure V2V Communications

*Note: Sub-titles are not captured in Xplore and should not be used

Hossam O. Ahmed
*College of Engineering and Technology
American University of the Middle East
Kuwait*
ORCID: 0000-0002-6825-9786

Hazem ElKanawati
*Computer Engineering
AASTMT, Sheraton Campus
Cairo, Egypt*
ORCID: 0009-0000-7437-8143

Mazen Moataz
*Computer Engineering
AASTMT, Sheraton Campus
Cairo, Egypt*
ORCID: 0009-0005-0783-3325

Abdelrhman Y. Abdelhamid
*Computer Engineering
AASTMT, Sheraton Campus
Cairo, Egypt*
ORCID: 0009-0007-4359-6114

Youssef Hany Shafie
*Computer Engineering
AASTMT, Sheraton Campus
Cairo, Egypt*
ORCID: 0009-0006-7761-5929

Rania Hussein
*Electrical and Computer Engineering
University of Washington
Washington, United States*
ORCID: 0000-0002-2859-9401

Abstract—Vehicle-to-vehicle (V2V) communication requires ultra-low latency, secure, and reliable authentication mechanisms to protect safety-critical message exchanges against spoofing and replay attacks. However, conventional multi-round cryptographic handshakes introduce prohibitive communication overhead and latency in high-density vehicular environments. This paper introduces CreV, a novel, lightweight Challenge–Response Exchange protocol and frame design engineered specifically to overcome this tradeoff without relying on specialized hardware architectures. At the core of the framework is a network-oriented protocol that leverages cryptographically secure, high-entropy pseudo-random challenges embedded directly within standard V2V broadcast structures. This frame-level embedding allows vehicle nodes to execute continuous verification natively within the data pipeline, eliminating heavy session-management overhead and minimizing channel congestion. We define the end-to-end networking architecture, detailing the optimized authentication frame formats inspired by IEEE 1609.2 and SAE J2735 standards, the verification pipelines, and token-refresh mechanics. Network-level emulation and architectural analysis demonstrate that CreV significantly reduces packet overhead while meeting the stringent real-time constraints of V2V networks, offering a highly scalable, low-overhead authentication paradigm for future intelligent transportation systems.

Index Terms—Challenge–response authentication; Frame design; Intelligent transportation systems; Lightweight cryptography; Low-latency authentication; Vehicle-to-vehicle (V2V) communication; Vehicular ad-hoc networks (VANETs).

I. INTRODUCTION

Intelligent Transportation Systems and Connected and Autonomous Vehicles have transitioned from conceptual frameworks to vital foundations of modern traffic management. At the core of these architectures lies Vehicle to Vehicle (V2V) communication, which enables vehicles to dynamically share

telemetry data such as velocity, global positioning, heading, and braking status [1], [2]. These safety critical transmissions, structured primarily around standardized protocols like SAE J2735 and IEEE 1609.2, demand ultra low latency and deterministic processing speeds to facilitate real time collision avoidance and automated maneuvering decisions [3].

However, because V2V channels inherently rely on open ad hoc wireless broadcasts, they are highly vulnerable to adversarial exploitation. Malicious actors can execute identity spoofing, message masquerading, or replay attacks by injecting stale, previously valid safety messages into the network to create artificial traffic disturbances or trigger catastrophic physical collisions. To mitigate these dangerous risks, V2V architectures mandate highly secure and reliable message authentication mechanisms capable of verifying the integrity and freshness of every inbound frame [4], [5].

Despite this clear necessity, existing cryptographic frameworks face severe deployment bottlenecks. Standard asymmetric Public Key Infrastructures and heavy multi round handshake protocols introduce substantial computational and communication overhead. In high density vehicular environments, such as packed urban intersections or crowded multi lane highways, the extra packet headers and repeated exchange rounds lead to channel congestion, increased packet drop rates, and unacceptable latency spikes. Conversely, proposed lightweight solutions frequently require specialized hardware elements like physical True Random Number Generators (TRNGs) or Field Programmable Gate Arrays (FPGAs), which complicate deployment logistics and increase manufacturing costs across diverse vehicle fleets [6]–[8].

To bridge this operational gap, this paper introduces *CreV*,

a novel lightweight Challenge Response Exchange protocol and frame design tailored explicitly for real time V2V communications. It is crucial to emphasize that the scope of this specific paper is dedicated entirely to introducing this novel conceptual architecture and evaluating its performance bounds through software simulation and network level emulation, while physical-channel modeling is left for future work. By embedding high entropy cryptographically secure pseudo random challenges directly into standard V2V broadcast payloads, neighboring vehicle nodes can execute continuous authentication natively within their active data pipelines. This approach minimizes session management state overhead while maintaining strict resistance against replay and spoofing vectors without imposing immediate hardware deployment requirements. Establishing the theoretical validity and structural efficiency of CreV via software modeling serves as a mandatory precursor to physical realization. Consequently, the next major milestone and future contribution following this study will focus explicitly on implementing and integrating this protocol onto a real hardware platform featuring a physical TRNG and an FPGA based system [9]–[11].

The primary contributions of this paper are summarized as follows:

- We introduce the conceptual architecture for CreV, a lightweight challenge response protocol designed to facilitate low overhead vehicular authentication via software emulated primitives.
- We propose an optimized network oriented frame structure that integrates security tokens seamlessly within layouts compliant with IEEE 1609.2 and SAE J2735 [3].
- We present a comprehensive network level emulation using a high entropy Python cryptographic framework to validate the execution flow of the verification pipeline, verifying its resilience to replay vectors and its structural efficiency under simulated conditions.

The remainder of this paper is organized as follows: Section II explores the preliminary background and protocol context regarding vehicular cryptographic standards; Section III delineates the proposed CreV protocol architecture and mechanics, including its specific frame configurations and operational logic; Section IV details the performance evaluation and experimental simulation results; and Section V concludes the paper with an overview of future research directions.

II. PRELIMINARY AND PROTOCOL CONTEXT

Securing vehicular environments remains a primary concern in intelligent transportation research. Early deployments relied heavily on Public Key Infrastructures to issue digital certificates for vehicle nodes. While public key cryptography provides strong non repudiation, the computational cost of asymmetric signature verification often conflicts with the strict real time constraints of vehicle safety networks. In dense traffic scenarios, processing a massive volume of incoming certificates introduces computational bottlenecks that can delay critical braking or collision warnings. To alleviate these computational demands, researchers proposed symmetric key

schemes, lightweight cryptographic primitives, and generic challenge response structures.

Crucially, existing literature in vehicular ad hoc networks has focused primarily on algorithmic efficiency, leaving a significant gap in entropy sourcing; specifically, no prior work has suggested utilizing physical TRNGs explicitly to secure V2V interactions. Recognizing this unaddressed security vector, our research pipeline is uniquely designed to pioneer the integration of a physical TRNG within the V2V broadcast ecosystem to radically boost cryptographic resilience against predictability attacks. This paper serves as the mandatory first step by establishing and validating the fundamental authentication framework via high fidelity software emulation. Building directly upon this simulated baseline, our immediate next milestone will involve physical hardware integration to implement a real TRNG based architecture that ensures hardware rooted security. Consequently, CreV initiates this paradigm by first establishing a streamlined, software emulated frame layout capable of seamless translation to physical hardware targets.

A. Supported Digital Signature Algorithms and Hash Relations

In strict compliance with the architectural framework, this standard supports the utilization of two robust digital signature algorithms for validating vehicle to vehicle messages. The first option is the Elliptic Curve Digital Signature Algorithm (ECDSA), which is explicitly specified in Federal Information Processing Standard (FIPS) 186 4. When executing ECDSA within this environment, the system optionally permits the inclusion of additional verification information inside the signature structure as detailed in SEC 1 Version 219.

The second supported option is the SM2 digital signature algorithm, which is formally specified in the ISO IEC 14888 3 standard. By providing native support for both ECDSA and SM2, the framework ensures high cryptographic flexibility, allowing intelligent transportation systems to adapt to diverse international security standards and regional regulatory mandates without requiring structural changes to the underlying frame design.

Within this framework, the hash algorithm associated with a signature algorithm is defined explicitly as the hash algorithm used to hash specific data structures that contain keys or signatures associated with that signature algorithm. For instance, the hash algorithm inherently coupled with SM2 over 256 bit curves is always SM3. Because SM2 does not support fast verification mechanics, the signature format for SM2 within this framework uses the EcsigP256Signature structure, which represents the signature parameter r strictly as an integer.

Furthermore, in alignment with IEEE Std 1609.2 2022, SM2 namespaces the hash function by prepending a specific identity string to the message prior to hashing. This identity string is mathematically defined as follows [3]:

$$ZA = H_{256}(ENTLA \parallel IDA \parallel a \parallel b \parallel x_G \parallel y_G \parallel x_A \parallel y_A) \quad (1)$$

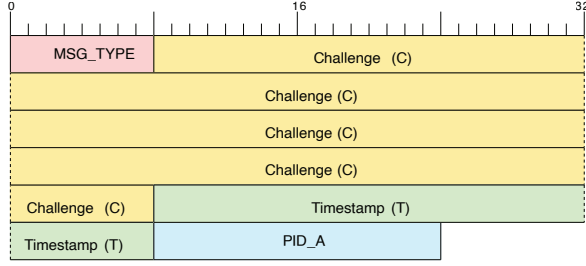


Fig. 1. Challenge Request Frame Layout (CHAL REQ).

The structural variables within this identity string equation are defined as follows according to the standard specifications:

- **ENTLA**: The two byte representation of the length in bits of the sender identity string.
- **IDA**: The unique identification string or identity of the transmitting sender node.
- **a and b**: The elliptic curve equation parameters, which are encoded strictly as unsigned integers of length 32 octets in network byte order.
- **xG and yG**: The specific x and y coordinates of the base generating point.
- **xA and yA**: The concrete x and y coordinates of the verification public key belonging to the sender, encoded as unsigned integers of length 32 octets in network byte order.

Once the identity string ZA is computed, the final digital signature is calculated over the combined hash input $H(ZA \parallel M)$ rather than executing the hash function over the message payload M alone. This structural design isolates the cryptographic context per vehicle identity, guaranteeing that the high entropy challenges remain securely bound to the authenticating nodes throughout the V2V broadcast pipeline.

B. Scalability Challenges and Computational Degradation

The execution of these advanced cryptographic operations introduces substantial scalability challenges regarding network data load and processing overhead. Because each broadcast packet must carry the multi byte identity strings, elliptic curve parameters, public keys, and the resulting digital signatures, the collective data requirement scales linearly with the frequency of safety message transmissions. As the density of the vehicular ad hoc network increases, the computational power required to verify these signatures degrades quadratically as a function of the number of surrounding cars [12]–[14].

In congested traffic environments where hundreds of vehicles continuously exchange telemetry data, the receiver nodes experience processing bottlenecks due to the sequential evaluation of the ZA identity string and the subsequent $H(ZA \parallel M)$ hash calculation for every single incoming frame. This degradation in processing efficiency directly risks exceeding the strict millisecond latency thresholds required for safety critical decisions, which underscores the necessity for

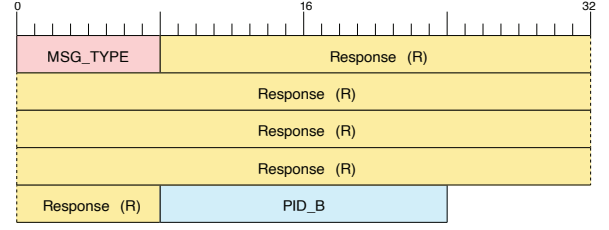


Fig. 2. Challenge Response Frame Layout (CHAL RESP).

the streamlined, low overhead frame design proposed in the CreV framework.

III. PROPOSED CREV PROTOCOL ARCHITECTURE AND MECHANICS

To overcome the severe latency overhead and computational degradation caused by processing the elliptic curve operations detailed in Equation 1, this paper introduces a streamlined, lightweight authentication mechanism. Instead of relying on time consuming public key identity string hashing and signature generation routines, the proposed framework executes verification natively within the data pipeline using efficient symmetric primitives. The proposed authentication packets were designed according to lightweight vehicular communication requirements and inspired by IEEE 1609.2 SPDU encapsulation concepts. Specifically, the bit level arrangements for initiating and answering an authentication phase are detailed in the Challenge Request Frame Layout shown in Fig. 1 and the Challenge Response Frame Layout shown in Fig. 2. The sequential execution flow and dynamic state transitions of the proposed protocol are formally mapped out in Algorithm 1.

A. Challenge Request Phase

The authentication procedure begins when Vehicle A initiates verification by generating a fresh 128 bit random challenge. To evaluate this behavior without physical hardware dependencies, the challenge generation process uses Python's `secrets` library to emulate cryptographically secure random number generation suitable for security oriented vehicular communication systems. This generated challenge is immediately combined with a temporal timestamp and a pseudonym identity to construct the complete authentication request payload. The challenge request packet is mathematically constructed according to the following formulation:

$$CHAL_REQ = MSG_TYPE \parallel C_A \parallel T_A \parallel PID_A \quad (2)$$

The architectural fields defined within Equation 2 comprise the following parameters:

- **MSG_TYPE**: A dedicated bit field that explicitly identifies the packet as an authentication request frame.
- **C_A** : The 128 bit high entropy pseudo random challenge generated by the simulated TRNG module.

- T_A : The timestamp freshness parameter utilized to guarantee network synchronicity and temporal validation.
- PID_A : The temporary pseudonym identifier of Vehicle A, ensuring privacy preservation across the broadcast domain.

B. Challenge Response Phase and Verification

Upon receiving the authentication request frame, Vehicle B first validates the freshness of the incoming timestamp T_A against its local clock to prevent malicious replay attacks. If the timestamp is valid, Vehicle B computes the authentication response using an HMAC SHA256 based authentication mechanism. The cryptographic response parameter R_B is derived as follows:

$$R_B = \text{HMAC}(K \parallel C_A \parallel PID_A \parallel PID_B \parallel \Gamma_{AB}) \quad (3)$$

K represents a securely pre-distributed symmetric key; key management is outside the scope of this study. The variable Γ_{AB} introduces a dynamic history tracking element to maximize security, defined as the output of an exclusive OR operation:

$$\Gamma_{AB} = T_A \oplus \Delta_{AB} \quad (4)$$

Here, Δ_{AB} represents a metric counting the exact number of successful authentication exchanges completed between these two specific vehicles within a predefined time window. This tracking mechanism introduces historical context to the key pipeline without storing heavy session states. Neighbor entries are maintained while communication is active and removed after a predefined inactivity period.

To maintain the compact frame structures necessary for congestion avoidance, the final generated message authentication code output is truncated to exactly 128 bits. This truncated value is encapsulated within the authentication response packet according to the following layout:

$$CHAL_RESP = MSG_TYPE \parallel R_B \parallel PID_B \quad (5)$$

where PID_B represents the pseudonym identifier of Vehicle B. Vehicle A subsequently recomputes the expected authentication response locally and executes a direct bitwise comparison against the received response parameter R_B . If both values match perfectly, the authentication succeeds and a secure communication session is established seamlessly.

IV. PERFORMANCE EVALUATION

A. Computational Overhead Evaluation

The protocol's mathematical overhead was benchmarked in Python 3.12.10 using an Intel Core i7-12700H processor with 16 GB DDR5 4800 MT/s memory. All experiments were executed strictly on a single-threaded CPU runtime to isolate the cryptographic execution latency. Standard ECDSA and SM2 operations were implemented according to the specifications defined in [3]. The benchmark was executed over 100 epochs,

Algorithm 1 CreV Rolling Broadcast Pipeline Execution

```

1: Initialization (Vehicle A):
2:  $\Delta_{AB} \leftarrow 0, \quad C_A \leftarrow \text{PRNG}(128)$ 
3: loop
4:    $T_A \leftarrow \text{Timestamp}()$ 
5:    $C_A \leftarrow \text{PRNG}(128)$ 
6:    $CHAL\_REQ \leftarrow \{MSG\_TYPE \parallel C_A \parallel T_A \parallel PID_A\}$ 
7:    $A \rightarrow * : CHAL\_REQ$ 
8:   On Intercepting  $CHAL\_REQ$  at Vehicle B:
9:      $\text{Verify}(T_A)$ 
10:     $\Gamma_{AB} \leftarrow T_A \oplus \Delta_{AB}$ 
11:     $R_B \leftarrow \text{HMAC}(K \parallel C_A \parallel PID_A \parallel PID_B \parallel \Gamma_{AB})$ 
12:     $CHAL\_RESP \leftarrow \{MSG\_TYPE \parallel R_B \parallel PID_B\}$ 
13:     $B \rightarrow * : CHAL\_RESP$ 
14:   On Intercepting  $CHAL\_RESP$  at Vehicle A:
15:     $\Gamma_{AB} \leftarrow T_A \oplus \Delta_{AB}$ 
16:    if  $R_B = \text{HMAC}(K \parallel C_A \parallel PID_A \parallel PID_B \parallel \Gamma_{AB})$  then
17:      Authentication Success
18:       $\Delta_{AB} \leftarrow \Delta_{AB} + 1$ 
19:    else
20:      Authentication Failure
21:    end if
22: end loop

```

Cryptographic Notation Index:

$CHAL_REQ$	Challenge request frame
$CHAL_RESP$	Challenge response frame
C_X	128 bit cryptographic challenge
R_X	Authentication response token
Γ_{AB}	Dynamic verification parameter
Δ_{AB}	Successful verification counter
$\text{HMAC}(\cdot)$	HMAC SHA256 authentication function
K	Shared symmetric secret key
PID_X	Pseudonym identity of Vehicle X
T_X	Timestamp freshness parameter
$\text{PRNG}(128)$	128 bit pseudo random generator
$\oplus$	XOR operator
$\parallel$	Concatenation operator
$*$	Broadcast domain indicator

where each epoch represents a complete execution cycle involving N vehicular nodes. The final measurements reported for each network scale correspond to the arithmetic average across all epochs. The evaluation uses simulated vehicle nodes and pairwise authentication exchanges to measure protocol-level computational overhead.

This evaluation models a scalability stress-test across varying vehicular densities. In decentralized V2V broadcast environments, each of the N vehicles transmits safety messages to surrounding nodes, producing a verification workload that scales quadratically as $O(N^2)$ under full broadcast reception assumptions. This verification matrix governs the overall trend observed in Fig. 3.

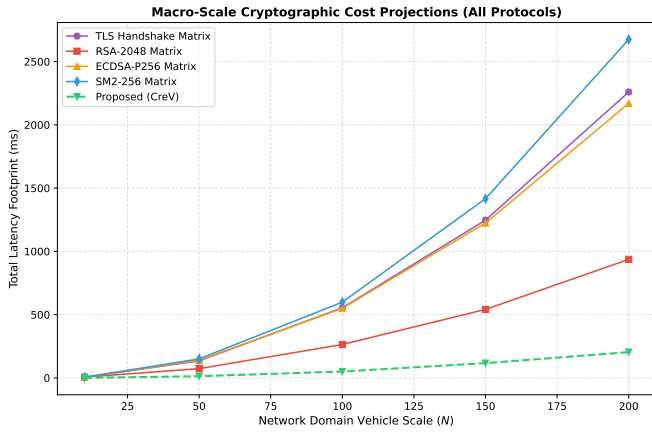


Fig. 3. Macro-scale cryptographic cost projections across increasing vehicular network density.

At low-density network conditions consisting of 10 vehicular nodes, the proposed CreV protocol achieved a cumulative computation overhead of only 0.430 ms, compared to 6.392 ms for RSA-2048, 5.450 ms for ECDSA-P256, and 5.811 ms for SM2-256. Under this baseline scenario, CreV reduced the total verification overhead by more than 92% relative to all evaluated protocols. As the vehicular density increased to 50 nodes, the computational gap increased. CreV maintained a comparatively low latency of only 12.618 ms, while RSA-2048 reached 74.552 ms, ECDSA-P256 reached 135.709 ms, and SM2-256 reached 147.136 ms. At the maximum density scale of 200 vehicles, CreV required 203.325 ms to complete the full verification matrix, while RSA-2048 exceeded 950 ms, ECDSA-P256 exceeded 2160 ms, and SM2-256 exceeded 2417 ms. As illustrated in Fig. 3, these results demonstrate that although all evaluated protocols follow the same quadratic verification scaling behavior, the proposed HMAC-based architecture maintains substantially lower cumulative latency as the vehicular density increases.

Signature generation scales linearly as $O(N)$, while signature verification scales quadratically as $O(N^2)$. Consequently, at higher network densities, the total latency becomes dominated by verification efficiency rather than signature generation overhead. RSA-2048 exhibits lower computational latency than ECDSA-P256 and SM2-256 because RSA verification relies on a comparatively lightweight public exponent operation. In contrast, ECDSA and SM2 require computationally intensive elliptic-curve scalar point multiplication during verification, resulting in substantially higher cumulative latency as the number of vehicles increases.

The implemented SM2 benchmark additionally complies with the protocol-level formatting and hashing requirements defined by the IEEE 1609.2-2022 standard [3]. Specifically, SM2 verification requires identity-bound preprocessing using the elliptic-curve parameters (a, b, x_G, y_G) together with the associated pre-hashing structure. This additional processing overhead accumulates across the verification matrix, causing

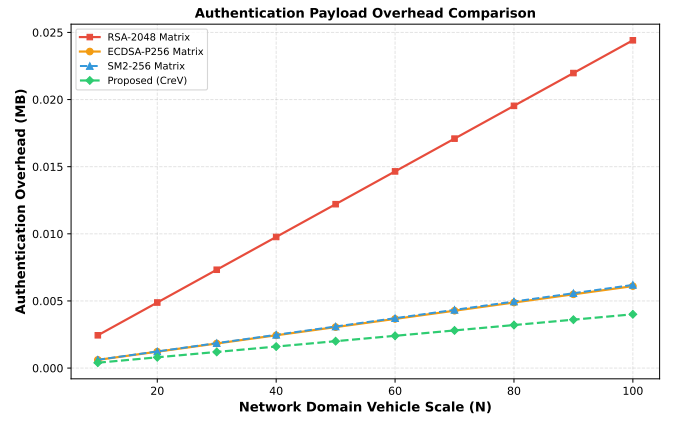


Fig. 4. Authentication communication overhead projections across varying vehicular densities.

SM2 to exhibit the highest aggregate latency among the evaluated asymmetric cryptographic schemes.

Although the proposed CreV protocol still operates under the same $O(N^2)$ verification scenario, its latency remains comparatively low on the millisecond scale. Unlike conventional public-key approaches, CreV replaces asymmetric verification mathematics with lightweight HMAC-SHA256 operations, which are computationally less expensive than elliptic-curve signature verification. This benchmark isolates the computational throughput advantages of the proposed framework independently from backend key-management architectures.

B. Authentication Communication Overhead

Figure 4 illustrates the authentication overhead transmitted by N vehicles. Authentication overhead was evaluated analytically using the number of added authentication bytes for each protocol across increasing vehicular densities. CreV shows smaller packet overhead since it uses a compact challenge-response frame structure, while also not involving asymmetric signatures. The authentication overhead added by CreV is 42 bytes, compared to 64 bytes for both ECDSA and SM2, 256 bytes for RSA, and kilobyte-scale overhead on TLS. Consequently, CreV exhibits a comparatively low authentication overhead. Although all schemes scale linearly in communication overhead, protocols with larger authentication payloads introduce higher cumulative network occupancy, while CreV maintains lower bandwidth consumption as vehicular density increases.

C. Replay Attack Detection Latency and Comparative Evaluation

The replay attack detection results are illustrated in Fig. 5. The attack model considers replay attacks at the protocol level. CreV imposes a remarkably small network footprint of only 42 bytes of protocol overhead. In stark contrast, alternative frameworks require extensive bandwidth due to their large key and signature sizes: ECDSA P256 demands 1160 bytes, SM2 256 requires 1224 bytes, TLS specifies

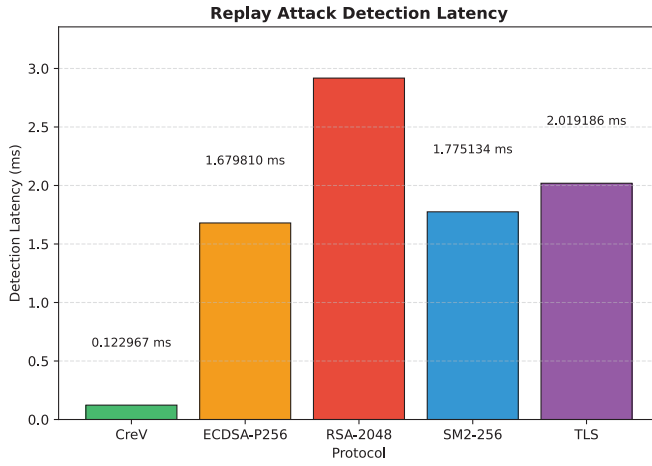


Fig. 5. Replay Attack Detection Latency Comparison of CreV, ECDSA-P256, RSA-2048, SM2-256, and TLS Protocols.

1408 bytes, and RSA 2048 enforces an overhead of 1544 bytes. By minimizing the packet size, CreV mitigates channel saturation within the shared wireless medium, successfully elevating the Packet Delivery Ratio to 7.0% at $N = 200$, whereas all other asymmetric frameworks stagnate completely at a lower delivery efficiency of 6.1%. Furthermore, when evaluating the specific domain of security defenses, CreV showcases a dedicated replay attack detection latency of just 0.123 ms, significantly outperforming the detection speeds of ECDSA P256 (1.680 ms), SM2 256 (1.775 ms), TLS (2.019 ms), and RSA 2048 (2.918 ms). This allows vehicle onboard units to detect and drop malicious frames almost instantaneously, providing a deterministic verification speed that satisfies the ultra low latency constraints necessary for safety critical intelligent transportation deployment.

V. CONCLUSION AND FUTURE WORK

This paper introduced CreV, a lightweight Challenge Response Exchange protocol for low latency vehicle to vehicle communication. By embedding high entropy pseudo random challenges within standard IEEE 1609.2 and SAE J2735 payloads, CreV eliminates time consuming multi round handshakes and heavy public key infrastructure overhead. Software simulations validated the protocol feasibility, proving its resistance to identity spoofing and replay attacks while minimizing packet size.

Since this study focused on establishing the foundational architecture via simulation, our next milestone will focus on physical implementation. Future work will also extend the security analysis to man-in-the-middle, relay, Sybil, denial-of-service, key-compromise, and counter-desynchronization attacks, as well as lost, delayed, and reordered packets, and

include comparison with PKI-based approaches. CreV will then be integrated onto a real hardware platform featuring a physical True Random Number Generator and an optimized FPGA based system to evaluate its deterministic processing speeds inside actual vehicular onboard units.

REFERENCES

- [1] G. Chai, H. Li, C. Li, N. Cheng, and F. R. Yu, "Data-Driven Robust Resource Allocation for Vehicular Communications via Deep Support Vector Clustering," *IEEE Transactions on Cognitive Communications and Networking*, vol. 12, pp. 6970–6987, 2026.
- [2] Y. Zhao, Z. Chen, M. Li, J. Zhang, G. Meng, Z. Chen, and H. Jiang, "Toward High-Efficient V2V Communication Networks: A Study on Channel Modeling and Level-Crossing Rate Analysis," *IEEE Transactions on Network Science and Engineering*, vol. 13, pp. 7949–7963, 2026.
- [3] "IEEE Standard for Wireless Access in Vehicular Environments—Security Services for Application and Management Messages," *IEEE Std 1609.2-2022 (Revision of IEEE Std 1609.2-2016)*, pp. 1–349, 2023.
- [4] O. Shmonin, V. Kuptsov, S. Trushkov, G. Serebryakov, and L. Li, "A Probabilistic Model for Vehicle-to-Vehicle Communication With Autonomous Resource Selection," *IEEE Open Journal of the Communications Society*, vol. 7, pp. 5039–5057, 2026.
- [5] W. He, J. Ye, L. Wang, X. Ge, and T. Han, "Learn to Solve Constrained Routing Problems in V2V Social Vehicle Networks," *IEEE Open Journal of the Communications Society*, vol. 7, pp. 4660–4680, 2026.
- [6] H. Mosanaei-Boorani and S. Bayat-Sarmadi, "A Digital Signature Architecture Suitable for V2V Applications," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 71, no. 2, pp. 731–739, 2024.
- [7] O. Sander, B. Glas, C. Roth, J. Becker, and K. D. Müller-Glaser, "Design of a Vehicle-to-Vehicle communication system on reconfigurable hardware," in *2009 International Conference on Field-Programmable Technology*, 2009, pp. 14–21.
- [8] L. Scripcariu, D. Burdia, and F. Diaconu, "FPGA Synthesis of an AES Encoder Circuit for Vehicular Communication Networks," in *2021 International Symposium on Signals, Circuits and Systems (ISSCS)*, 2021, pp. 1–4.
- [9] H. O. Ahmed, S. Redif, and Y. Li, "TeLo-ROPN TRNG: FPGA-Based True Random Number Generation Using Tent and Logistic Chaotic Maps With Ring Oscillator Array," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 73, no. 5, pp. 3532–3545, 2026.
- [10] H. O. Ahmed, "MCLM-BXN: Design of a High-Entropy TRNG Using Modified Chaotic Logistic Maps and Braided XOR Networks on FPGA," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 45, no. 4, pp. 2041–2054, 2026.
- [11] H. O. Ahmed, D. Kim, and W. J. Buchanan, "A True Random Number Generator Based on Race Hazard and Jitter of Braided and Cross-Coupled Logic Gates Using FPGA," *IEEE Access*, vol. 12, pp. 182 943–182 955, 2024.
- [12] T. Qayyum, A. Tariq, I. Taleb, M. A. Serhani, and Z. Trabelsi, "A performance-optimized V2V task offloading framework for real-time vehicular communication," *Scientific Reports*, vol. 16, no. 1, p. 14587, Mar. 2026. [Online]. Available: <https://doi.org/10.1038/s41598-026-44686-z>
- [13] G. Shah, M. Zaman, M. Saifuddin, B. Toghi, and Y. Fallah, "Scalable Cellular V2X Solutions: Large-Scale Deployment Challenges of Connected Vehicle Safety Networks," *Automotive Innovation*, vol. 7, no. 3, pp. 373–382, Aug. 2024. [Online]. Available: <https://doi.org/10.1007/s42154-023-00277-6>
- [14] W. Ahmad, F. Ali, and A. Ali, "Scalable AI-driven intent-based vehicle-to-everything communications for secure, low-latency, and energy-efficient smart cities," *Vehicular Communications*, vol. 56, p. 100973, 2025. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2214209625001007>