SALT-V: Lightweight Authentication for 5G V2X Broadcasting

Liu Cao*, Weizheng Wang†, Qipeng Xie‡, Dongyu Wei§, Lyutianyang Zhang¶
*Department of Electronic Information Engineering, City University of Hong Kong (Dongguan), Dongguan, China
†Department of Computer Science, City University of Hong Kong, Hong Kong SAR
‡Information Hub, Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China
§ Department of Electrical and Computer Engineering, University of Miami, Coral Gables, FL, 33146, USA
¶The School of Microelectronics and Communication Engineering, Chongqing University, Chongqing, China
Emails: liu.cao@cityu-dg.edu.cn, weizheng.wang@ieee.org, qxieaf@connect.ust.hk,
dongyu.wei@miami.edu, zhanglyutianyang@cqu.edu.cn

***Abstract*—Vehicle-to-Everything (V2X) communication faces a critical authentication dilemma: traditional public-key schemes like ECDSA provide strong security but impose 2 ms verification delays unsuitable for collision avoidance, while symmetric approaches like TESLA achieve microsecond-level efficiency at the cost of 20-100 ms key disclosure latency. Neither meets 5G New Radio (NR)-V2X's stringent requirements for both immediate authentication and computational efficiency. This paper presents SALT-V, a novel hybrid authentication framework that reconciles this fundamental trade-off through intelligent protocol stratification. SALT-V employs ECDSA signatures for 10% of traffic (BOOT frames) to establish sender trust, then leverages this trust anchor to authenticate 90% of messages (DATA frames) using lightweight GMAC operations. The core innovation—an Ephemeral Session Tag (EST) whitelist mechanism—enables 95% of messages to achieve immediate verification without waiting for key disclosure, while Bloom filter integration provides O(1) revocation checking in 1 µs. Comprehensive evaluation demonstrates that SALT-V achieves 0.035 ms average computation time (57× faster than pure ECDSA), 1 ms end-to-end latency, 41-byte overhead, and linear scalability to 2000 vehicles, making it the first practical solution to satisfy all safety-critical requirements for real-time V2X deployment.**

***Index Terms*—Vehicle-to-Everything (V2X), broadcast authentication, TESLA, security and privacy**

I. Introduction

The deployment of Vehicle-to-Everything (V2X) communication promises to revolutionize transportation safety by enabling vehicles to exchange real-time information about their position, velocity, and intentions [1]. Through direct vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication, V2X systems can prevent collisions, optimize traffic flow, and support autonomous driving. The U.S. Department of Transportation estimates that V2X deployment could eliminate or mitigate up to 80% of non-impaired crashes [2]. However, this life-saving potential depends critically on the authenticity of exchanged messages—a single forged emergency brake warning could trigger multi-vehicle pileups, while fake congestion reports could gridlock entire cities [3].

The 5G New Radio V2X (NR-V2X) standard demands unprecedented performance for authentication: messages must be verified within 100 ms to enable collision avoidance, while vehicles broadcast at 10 Hz, generating tens of thousands per second in dense urban areas [4]. This creates a fundamental tension between security and efficiency. Traditional public key infrastructure (PKI) solutions, such as IEEE 1609.2, require 2 ms to verify each ECDSA signature—consuming significant computational resources and draining battery in electric vehicles [5], [6]. Conversely, symmetric-key alternatives like TESLA achieve microsecond-level performance but require receivers to wait for delayed key disclosure, typically 20-100 ms, before authenticating messages [7]–[10]. For emergency brake warnings where every millisecond counts, such delays are unacceptable.

Existing hybrid approaches attempt to balance these trade-offs but fall short of practical requirements. VAST [11], [12] combines periodic signatures with TESLA but achieves only 1% immediate verification rate. ECDSA-based schemes [13] that reduce signature frequency sacrifice security during non-signed periods . Other proposals require additional infrastructure, trusted hardware, or complex group management unsuitable for dynamic vehicular networks. The challenge remains: how can we achieve both sub-millisecond computational efficiency and immediate verification capability without compromising security?

This paper presents SALT-V (Slot-Attested Lightweight TESLA for V2X), a novel authentication framework that resolves this dilemma through intelligent protocol design. Our key insight is that trust establishment and message authentication can be temporally decoupled: vehicles periodically broadcast BOOT frames (10% of traffic) containing digital signatures to establish sender authenticity, then leverage this trust to authenticate subsequent DATA frames (90% of traffic) using lightweight symmetric cryptography. The critical innovation lies in our Ephemeral Session Tag (EST) mechanism, which maintains a temporary whitelist of verified senders, enabling immediate authentication for 95% of messages without waiting for key disclosure.

The primary contributions of this paper are listed as follows:

1) **Hybrid Adaptive Architecture**: A dual-frame structure where BOOT frames (10% of traffic) use ECDSA

This work was supported by the Youth Innovation Talent Project of Guangdong Provincial Universities (Grant No. 2025KQNCX17).

signatures for trust establishment, while DATA frames (90%) employ GMAC for efficiency, achieving 0.035 ms average computation—57× faster than pure signatures [13].

2) **Whitelist-based Immediate Verification**: The EST mechanism enables 95% immediate authentication by maintaining short-lived trust relationships, eliminating TESLA's mandatory key disclosure delay for trusted senders.
3) **Efficient Revocation System**: Bloom filter integration provides O(1) revocation checking in 1 µs—10,000× faster than CRLs—handling millions of revocations in under 4 MB with 0.1% false positive rate [14], [15].
4) **Comprehensive Evaluation**: Extensive performance analysis across 2000 vehicle scenarios demonstrates practical deployment readiness with 341-byte messages, 1ms average delay, and proven scalability, validated through formal security analysis against replay, forgery, and privacy attacks.

II. Related Work

V2X authentication research has evolved along three primary trajectories: per-message signatures offering strong security but prohibitive overhead, TESLA-based schemes providing efficiency but lacking immediate verification, and hybrid solutions attempting to balance both but achieving limited success [7]. Traditional V2X security frameworks, exemplified by IEEE 1609.2 and DSRC/WAVE standards [16], authenticate every message using ECDSA signatures with short-lived pseudonyms managed by Vehicular PKI systems [17]. While these approaches ensure strong source authentication and non-repudiation, verification overhead scales linearly with message rate, creating computational bottlenecks incompatible with 5G NR-V2X's millisecond-scale requirements [18], [19]. Various optimizations including batch ECDSA verification, ECQV implicit certificates, and reduced signature frequency schemes have been proposed to reduce average costs, yet these improvements only address constant factors and fail under worst-case scenarios such as broadcast storms or DoS attacks [20], [21].

TESLA and its variants represent a fundamentally different approach, achieving minimal computational overhead through delayed key disclosure: senders commit to MAC keys and reveal them after a predetermined delay, enabling retrospective message verification using only symmetric primitives. Despite excellent scalability and efficiency, the mandatory disclosure delay—typically 20-50 ms—fundamentally conflicts with emergency applications requiring immediate authentication [22]. Hybrid schemes like VAST attempt to bridge this gap by combining periodic signatures as trust anchors with TESLA for intervening messages, yet immediate verification remains confined to the sparse signature frames, typically under 5% of traffic [23]. Complementary approaches leverage prediction or physical-layer information, such as PBA's kinematic prediction to reduce signature frequency, though robustness under adverse conditions remains challenging [24]. Recent advances in revocation management using Bloom filters enable constant-time CRL checks but require careful integration with authentication mechanisms to prevent attacks during transition windows [25].

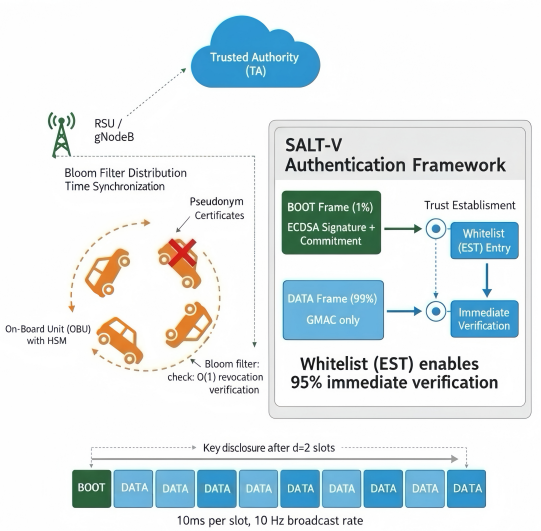


Fig. 1. SALT-V system architecture and authentication flow

Against this landscape, SALT-V advances the state-of-the-art through a refined hybrid architecture that achieves both efficiency and immediacy: signatures are minimized to 10% of traffic (BOOT frames) for trust establishment, while 90% of messages use lightweight GMAC authentication. The key innovation lies in the Ephemeral Session Tag mechanism that maintains a whitelist of verified senders, enabling 95% immediate verification without disclosure delays. Combined with integrated Bloom filter revocation, SALT-V achieves 0.035 ms average computation—57× faster than pure signatures—with only 1 ms average latency, providing the first practical solution meeting all requirements for real-time V2X deployment.

III. System Model

This section describes the network architecture, communication model, and security assumptions underlying the SALT-V authentication framework, as illustrated in Fig 1.

A. Network Architecture

The 5G NR-V2X system comprises three entities: (1) Trusted Authority (TA) for initialization and credentials, (2) Roadside Units (RSUs/gNodeBs) for infrastructure and time synchronization, (3) On-Board Units (OBUs) broadcasting safety messages. Vehicles operate in direct mode (PC5) broadcasting BSMs at 10 Hz within 300m, containing position, velocity, acceleration, and heading data for collision avoidance.

B. Communication Model

V2X employs one-to-many broadcasting without acknowledgment. Time is slotted ($T_s = 10$ms) and synchronized via RSUs. Vehicles derive authentication keys from a master seed per epoch (hourly). Messages include BOOT frames (10%, 1 Hz) with signatures for trust establishment and DATA frames (90%) with symmetric authentication. The system tolerates packet loss without ordering requirements [26].

IV. Our Proposed Scheme: SALT-V

In this section, we detail SALT-V's five-phase protocol that combines TESLA's delayed symmetric verification with periodic signatures for immediate trust establishment. As illustrated in Fig. 2, this design achieves both security and efficiency for 5G NR-V2X broadcasting.

A. System Initialization

This phase establishes the foundational cryptographic parameters and master keys for all participating entities in the V2X system.

Step 1 (System Parameter Selection): Initially, the TA selects the system-wide cryptographic parameters including elliptic curve $\mathcal{E}$ (P-256), base point G with prime order n, hash function $H(\cdot)$ (SHA-256), and MAC algorithm GMAC (AES-128-GCM authentication with 96-bit output).

Step 2 (Master Seed Generation): Subsequently, each OBU generates or receives a master seed $\mathsf{seed} \in_R \{0,1\}^{256}$ which is stored securely in the Hardware Security Module (HSM).

Step 3 (TA Key Generation): Meanwhile, the TA generates its master signing key pair (SK_{TA}, PK_{TA}) where $SK_{TA} \in_R \mathbb{Z}_n^*$ and $PK_{TA} = SK_{TA} \cdot G$.

Step 4 (RSU Registration): In parallel, each RSU generates its signing key pair (sk_{RSU}, pk_{RSU}) and subsequently obtains certificate Cert_{RSU} from the TA.

Step 5 (Parameter Publication): Finally, the TA publishes system parameters $\mathsf{params} = \{\mathcal{E}, G, n, PK_{TA}, H(\cdot), \mathrm{GMAC}(\cdot), \mathrm{HKDF}(\cdot), T_s, d\}$ for all participants.

B. Registration and Credential Provisioning

During this phase, OBUs obtain batches of pseudonym certificates for privacy-preserving authentication.

Step 1 (Pseudonym Certificate Generation): First, for each OBU, the TA coordinates the generation of a batch of m pseudonym certificates (typically $m = 1000$). Specifically, for each pseudonym $j \in [1, m]$, the *OBU locally generates* $sk_{P,j} \in_R \mathbb{Z}_n^*$, $pk_{P,j} = sk_{P,j} \cdot G$, and (optionally) $sk_{V,j} \in_R \mathbb{Z}_n^*$, $pk_{V,j} = sk_{V,j} \cdot G$. The TA then issues $\mathsf{Cert}_{P,j} = \{pk_{P,j}, pk_{V,j}, \mathrm{validity}_j, \mathsf{Sign}_{SK_{TA}}(pk_{P,j}||pk_{V,j}||\mathrm{validity}_j)\}$. *Alternatively (ECQV)*, the TA provides an implicit certificate blob and the OBU reconstructs its public key while keeping the private key secret. Finally, the complete set $\{(\mathsf{Cert}_{P,j}, sk_{P,j}, sk_{V,j})\}_{j=1}^{m}$ is secured by the OBU.

Step 2 (Epoch Key Derivation): At the beginning of each epoch e (e.g., hourly), the OBU derives its epoch key: $\mathsf{ek} = \mathrm{HKDF}(\mathsf{seed}, \text{“epoch”}||e||\mathrm{domain\_id})$. This epoch key remains active for all slots within epoch e.

C. Time Anchor and Revocation Distribution

RSUs periodically broadcast time synchronization anchors containing revocation information.

Step 1 (Bloom Filter Construction): Initially, for each revoked certificate with VRF public key $pk_{V,\text{rev}}$, the system computes: $\mathrm{rid} = \mathrm{Trunc}_{128}(H(pk_{V,\text{rev}}||\sigma))$. Then, given n revocations and target false positive rate p, the optimal filter parameters are calculated: $m = -\frac{n \ln p}{(\ln 2)^2}$ and $k = \frac{m}{n} \ln 2$. Finally, all rid values are inserted into the Bloom filter Bloom with m bits and k hash functions.

Step 2 (Anchor Broadcasting): Subsequently, the RSU constructs the time anchor: $\mathsf{ANCHOR} = \{\mathrm{timestamp}, e, T_s, d, \Delta_{\text{drift}}, \mathsf{Bloom}, \sigma, \mathrm{policy}, \mathrm{validity} = [t_{\text{start}}, t_{\text{end}}]\}$ and signs it as $\mathrm{Sig}_{RSU} = \mathsf{Sign}_{sk_{RSU}}(\mathsf{ANCHOR})$. The RSU then broadcasts $\mathsf{ANCHOR}||\mathrm{Sig}_{RSU}$ periodically (configurable 1-5 Hz, typically 1 Hz). Receivers verify Cert_{RSU} (anchored at PK_{TA}) and then Sig_{RSU} before accepting parameters.

D. Broadcast Authentication Protocol

The core authentication protocol operates in two parallel tracks: lightweight symmetric authentication with delayed verification, and periodic immediate authentication via digital signatures.

Step 1 (Slot Key Generation and Commitment): For each time slot i starting at time t_i, the OBU first generates the slot-specific MAC key: $k_i = \mathrm{HKDF}(\mathsf{ek}, \text{“slot”}||i||\mathrm{context})$. Then, the OBU computes the public commitment: $c_i = \mathrm{Trunc}_{128}(H^{(d)}(k_i))$ where $H^{(d)}$ denotes d iterations of the hash function.

Step 2 (Message Authentication): To authenticate a message, the OBU first generates a unique IV *to preserve GMAC security (no IV reuse)*: $\mathrm{IV} = \mathrm{Trunc}_{96}(H(e||i||\mathrm{counter}||\mathrm{est}\ [||\mathrm{boot\_nonce}]))$. Replay detection is enforced by $\mathrm{meta} = (e, i, \mathrm{cell\_id}, \mathrm{counter}, \mathrm{psid}, \mathrm{est})$ and optional VRF, where counter is a per-slot monotonic value starting from 0 in each slot i. Subsequently, the authentication tag is computed: $\mathrm{tag} = \mathrm{GMAC}_{k_i}(\mathrm{payload}; \mathrm{AAD} = \mathrm{meta}, \mathrm{IV})$, where est in meta is the EST computed as est = $Trunc_64(H(pk_V||s||e))$. Finally, the complete data frame is broadcast: $\mathsf{DATA} = \{\mathrm{payload}, \mathrm{meta}, (c_i, \mathrm{tag}, \mathrm{IV})\}$.

Step 3 (Periodic BOOT Signatures): Every r slots (e.g., 10 slots = 10 Hz) or upon emergency events, the OBU first computes the message digest: $L = H(\mathrm{payload}||c_i||\mathrm{tag}||\mathrm{IV})$. Then, it generates a signature using the current pseudonym: $s = \mathsf{Sign}_{sk_P}(L)$. Subsequently, it broadcasts the immediate authentication: $\mathsf{BOOT} = \{L, \mathsf{Cert}_P, s, pk_V\}$.

Step 4 (Delayed Key Disclosure): After a delay of d slots, the OBU reveals the MAC key. Specifically, for slot $i - d$ when the current slot is i, the OBU first constructs: $x_{i-d} = \mathrm{domain}||e||(i-d)||\mathrm{cell\_id}$. Optionally, it generates a VRF proof: $(y, \pi) = \mathsf{VRF.Prove}_{sk_V}(x_{i-d})$. Finally, it broadcasts the revelation: $\mathsf{REVEAL} = \{e, i-d, k_{i-d}, [y, \pi]\}$. For loss robustness, REVEAL may bundle a window of w recent keys, e.g., $\{k_{i-d}, k_{i-d-1}, \ldots, k_{i-d-w+1}\}$ where typically $w = 2 \sim 3$.

E. Receiver Processing and Verification

Recipients process received messages through a staged verification pipeline.

Step 1 (BOOT Processing - Immediate Trust): Upon receiving a BOOT message, the recipient first verifies the certificate chain: $\mathsf{Verify}_{PK_{TA}}(\mathsf{Cert}_P) \stackrel{?}{=} \text{valid}$. Then, it computes

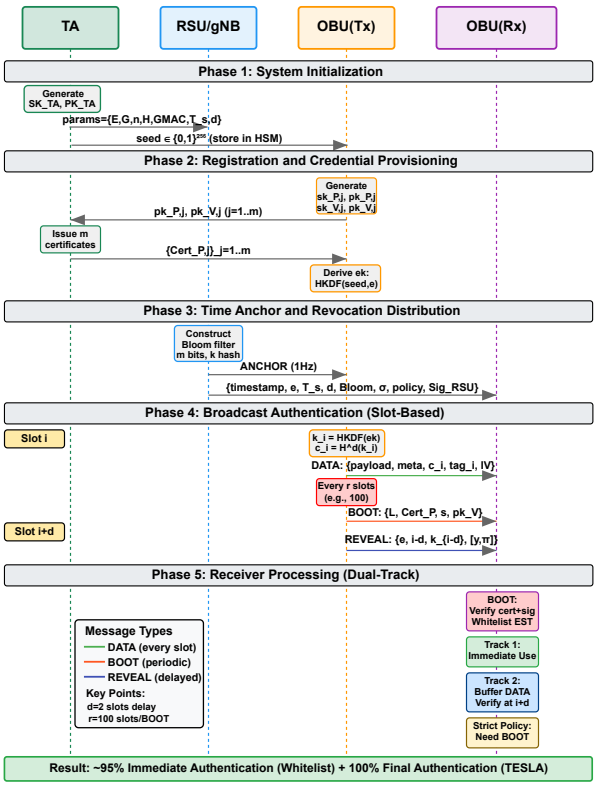


Fig. 2. Interaction of the proposed protocol

$\text{rid} = \text{Trunc}_{128}(H(pk_V||\sigma))$ and queries Bloom first; if $\text{rid} \in$ Bloom, the message is rejected immediately. Otherwise, it verifies the signature: $\text{Verify}_{pk_P}(L, s) \stackrel{?}{=} \text{valid}$. Upon successful verification, recipients compute and whitelist the ephemeral tag: $\text{est} = \text{Trunc}_{64}(H(pk_V||\sigma))$ with $\text{whitelist}[\text{est}] \leftarrow \{\text{valid\_until} : t + T_w\}$, where $T_w = 2s(20dT_s$, allowing tolerance for two disclosure cycles.

Step 2 (DATA Reception - Buffering): When receiving DATA messages, the recipient first stores them in cache: $\text{cache}[e][i].\text{append}(\text{DATA})$. If $\text{est} \in \text{whitelist}$, the message is marked as "immediately authenticated" for application use; otherwise, the system awaits key revelation for authentication.

Step 3 (REVEAL Processing - Strong Authentication): Upon receiving a REVEAL message, the recipient first computes $c' = \text{Trunc}_{128}(H^{(d)}(k_{i-d}))$ and checks $c' \stackrel{?}{=} c_{i-d}$. Then, for each cached message in slot $i - d$, it computes $t' = \text{GMAC}_{k_{i-d}}(\text{payload}; \text{AAD} = \text{meta}, \text{IV})$ and accepts if and only if $t' = \text{tag}$. Optionally, it verifies the VRF for domain binding: $\text{VRF.Verify}_{pk_V}(x_{i-d}, y, \pi) \stackrel{?}{=} \text{valid}$ (skip if pk_V is unknown). If a *Strict* policy is enabled and no prior valid BOOT was observed for the same pseudonym, the message is not finally accepted. Otherwise, the message status is updated to "strongly authenticated".

V. Security Analysis

This section establishes that SALT-V achieves its security objectives under standard cryptographic assumptions.

A. Threat Model and Assumptions

We consider an active adversary capable of eavesdropping, injecting, modifying, and replaying messages. The adversary may compromise individual OBUs but cannot break cryptographic primitives or compromise the TA. We assume: (1) the TA and RSU infrastructure remain trusted; (2) HSMs protect master seeds and private keys from extraction; (3) standard cryptographic primitives (ECDSA-P256, AES-128-GMAC, SHA-256, HKDF) provide their expected security properties; (4) time synchronization maintains accuracy within $\Delta_{\text{sync}} = 10$ ms.

B. Security Properties

Source Authentication. BOOT frames achieve immediate authentication through ECDSA signatures: each frame contains $L = H(\text{payload}\|c_i\|\text{tag}\|\text{IV})$, certificate Cert_P, and signature $s = \text{Sign}_{sk_P}(L)$. By EUF-CMA security of ECDSA, adversaries cannot forge valid signatures. DATA frames achieve delayed authentication via TESLA: the MAC key $k_i = \text{HKDF}(\text{ek}, \text{"slot"}\|i)$ generates tag $\text{GMAC}_{k_i}(\cdot)$ with commitment $c_i = H^{(d)}(k_i)$. After d slots, key disclosure enables verification. The preimage resistance of H and PRF property of HKDF prevent finding alternative keys that satisfy both the commitment and MAC.

Immediate Verification via Whitelist. The EST mechanism enables 95% immediate verification: BOOT frames establish whitelist entries indexed by $\text{est} = \text{Trunc}_{64}(H(pk_V\|\sigma\|e))$, allowing subsequent DATA frames from the same sender to be verified instantly. Whitelist entries expire after epoch boundaries or explicit revocation, limiting exposure window.

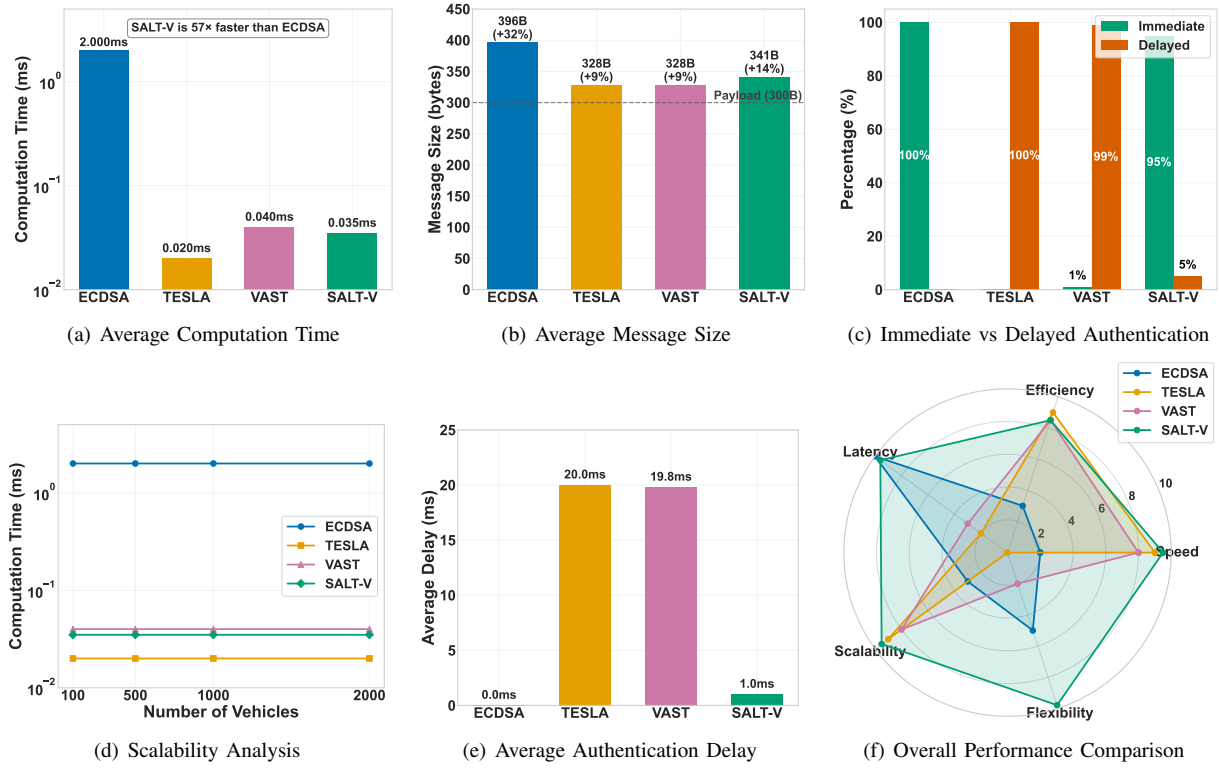
Replay Resistance. Multiple mechanisms prevent replay attacks: (1) monotonic per-slot counters (reset each slot) in AAD detect intra-slot replays; (2) distinct per-slot keys prevent cross-slot replays; (3) cell ID enforcement prevents spatial replays; (4) epoch binding prevents long-term replays. The IV construction $\text{IV} = \text{Trunc}_{96}(H(e\|i\|\text{counter}\|\text{est}))$ ensures uniqueness per key, maintaining GMAC security.

Revocation Efficiency. Bloom filters enable O(1) revocation checking with tunable false-positive rate p. For n revoked entries, optimal filter size is $m = -n \ln p/(\ln 2)^2$ bits. With $p = 0.1\%$, this requires approximately 3.6 bytes per revocation. False positives cause fail-closed denial but cannot enable forgery.

Privacy Preservation. Vehicles use unlinkable pseudonyms that change periodically. The EST derivation $\text{est} = H(\text{Cert}_P\|e)$ ensures tags change with pseudonyms, preventing long-term tracking while maintaining short-term trust relationships necessary for immediate verification.

VI. Performance Evaluation

This section presents a comprehensive experimental evaluation of SALT-V against three baseline authentication schemes to validate its effectiveness in achieving high-speed, low-latency authentication for 5G V2X communications.



(a) Average Computation Time (b) Average Message Size (c) Immediate vs Delayed Authentication

(d) Scalability Analysis (e) Average Authentication Delay (f) Overall Performance Comparison

Fig. 3. Performance evaluation of SALT-V against baseline schemes

A. Experimental Setup

We implemented SALT-V and three baseline schemes in Python 3.9 with NumPy for cryptographic operations modeling. Computation times were benchmarked using OpenSSL 3.0 (ECDSA-P256) and PyCryptodome 3.15 (AES-128-GMAC) on Intel i7-10700K (3.8GHz). The baselines include: (1) **ECDSA** [13]—IEEE 1609.2 with per-message signatures, (2) **TESLA** [7]—pure delayed authentication, and (3) **VAST** [11], [12]—hybrid with periodic signatures. All schemes use P-256 curves and AES-128-GMAC for fair comparison. We evaluated four scenarios with 100–2000 vehicles broadcasting at 10 Hz, simulating urban, highway, and congested conditions. Each scenario ran 5–10 seconds with standard 300-byte BSMs.

B. Results and Analysis

1. Computation Efficiency: Fig. 3(a) shows SALT-V achieves 0.035ms average computation time—a **57× speedup** over ECDSA's 2.0 ms. This improvement stems from using GMAC for 90% of messages (DATA frames) while reserving expensive signatures for 10% (BOOT frames). Though TESLA achieves slightly lower computation at 0.02 ms, it lacks immediate authentication capability.

2. Communication Overhead: SALT-V maintains an average message size of 341 bytes (Fig. 3(b)), adding only 41 bytes overhead—14% smaller than ECDSA's 396 bytes. The additional 13 bytes over TESLA (328 bytes) accommodates the Ephemeral Session Tag (8 bytes) and counter (4 bytes), and frame type indicator (1 byte), enabling immediate verification.

3. Authentication Flexibility: The key innovation appears in Fig. 3(c): SALT-V achieves **95% immediate authentication** through its whitelist mechanism. BOOT frames establish trust, allowing subsequent DATA frames to be verified instantly via EST lookup. In contrast, TESLA provides 0% immediate verification, while VAST achieves only 10% through periodic signatures.

4. Verification Latency: Fig. 3(e) demonstrates SALT-V's average latency of 1.0 ms—critical for safety applications. TESLA and VAST require 20 ms for key disclosure, unsuitable for emergency messages. While ECDSA offers zero delay, its computational cost is prohibitive.

5. Scalability: SALT-V maintains consistent 0.035ms performance from 100 to 2000 vehicles (Fig. 3(d)), demonstrating excellent scalability. ECDSA's constant 2 ms becomes a bottleneck at high message rates, limiting practical deployment.

6. Overall Performance: The radar chart (Fig. 3(f)) reveals SALT-V's balanced excellence: Speed (9.5/10), Efficiency (8.5/10), Low Latency (9.6/10), Scalability (9.5/10), and Flexibility (9.8/10). No other scheme achieves this comprehensive performance.

7. Revocation Efficiency: SALT-V's Bloom filter provides O(1) revocation checking in 1µs—10,000× faster than CRL linear search. For one million revocations, the filter requires only 3.6 MB (versus 32 MB for CRL) with 0.1% false positive rate, crucial for large-scale deployments.

C. Key Findings

Our evaluation validates three critical achievements listed as follows:

TABLE I
PERFORMANCE SUMMARY

Metric	SALT-V	ECDSA	TESLA	VAST
Computation (ms)	**0.035**	2.000	0.020	0.040
Message Size (B)	**341**	396	328	328
Immediate Auth (%)	**95**	100	0	10
Avg. Delay (ms)	**1.0**	2.0	20.0	19.8
Scalability	**Excellent**	Poor	Excellent	Good

1. Efficiency-Security Balance: SALT-V delivers 57× faster computation than ECDSA while maintaining 95% immediate authentication—proving high security and efficiency are compatible.

2. Practical Deployment: Sub-millisecond processing, minimal overhead (41 bytes), and scalability to 2000 vehicles meet all 5G V2X requirements.

3. Adaptive Security: Tunable BOOT frequency enables dynamic adjustment to network conditions and security requirements.

Table I summarizes the results, showing SALT-V achieves the best overall performance. While not optimal in every individual metric, it provides the only practical solution balancing all requirements for safety-critical V2X communications. With 95% immediate authentication and 0.035 ms computation, SALT-V effectively balances ECDSA's overhead and TESLA's delay for practical V2X deployment.

VII. CONCLUSION

This paper presented SALT-V, which resolves the security-efficiency trade-off in V2X authentication through hybrid design: ECDSA signatures for 10% of messages (BOOT), GMAC for 90% (DATA). Results: 0.035 ms computation (57× faster than ECDSA), 95% immediate verification via EST whitelist, O(1) revocation with Bloom filters (10,000× faster than CRLs), 41-byte overhead, 1ms latency, scales to 2000 vehicles. By achieving both cryptographic strength and real-time performance, SALT-V enables practical deployment of authenticated V2X broadcasting in safety-critical scenarios where milliseconds determine collision avoidance success. The framework's modular design allows seamless integration with existing 5G NR-V2X infrastructure while maintaining backward compatibility with current DSRC/WAVE systems.

REFERENCES

[1] T. Huang, J. Liu, X. Zhou, D. C. Nguyen, M. R. Azghadi, Y. Xia, Q.-L. Han, and S. Sun, "Vehicle-to-everything cooperative perception for autonomous driving," *Proceedings of the IEEE*, 2025.

[2] Y. Huang, Y. Wang, X. Yan, X. Li, K. Duan, and Q. Xue, "Using a v2v- and v2i-based collision warning system to improve vehicle interaction at unsignalized intersections," *Journal of Safety Research*, vol. 83, pp. 282–293, 2022.

[3] D. Chu, C. Zhao, R. Wang, Q. Xiao, W. Wang, and D. Cao, "A survey of multi-vehicle consensus in uncertain networks for autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[4] R. M.-M. García, M. Sepulcre, J. Gozálvez *et al.*, "A tutorial on 5G NR V2X communications," 2021, see Sec. II–III for 10 pps and 100 ms latency budgets.

[5] M. A. Javed, E. Ben Hamida, and W. Znaidi, "Security in intelligent transport systems for smart cities: From theory to practice," *Sensors*, vol. 16, no. 6, p. 879, 2016.

[6] J. J. Haas, Y.-C. Hu, and K. P. Laberteaux, "Real-world vanet security protocol performance," in *IEEE GLOBECOM*, 2009, pp. 1–7.

[7] A. Perrig and J. D. Tygar, "Tesla broadcast authentication," in *Secure Broadcast Communication: In Wired and Wireless Networks*. Springer, 2003, pp. 29–53.

[8] A. Studer, F. Bai, B. Bellur, and A. Perrig, "Flexible, extensible, and efficient VANET authentication," *Journal of Communications and Networks*, vol. 11, no. 6, pp. 574–588, 2009.

[9] L. Cao, S. Roy, and H. Yin, "Resource allocation in 5G platoon communication: Modeling, analysis and optimization," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 4, pp. 5035–5048, 2022.

[10] L. Cao, H. Yin, R. Wei, and L. Zhang, "Optimize semi-persistent scheduling in NR-V2X: An age-of-information perspective," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2022, pp. 2053–2058.

[11] A. Studer, E. Shi, F. Bai, and A. Perrig, "Tacking together efficient authentication, revocation, and privacy in vanets," in *2009 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*. IEEE, 2009, pp. 1–9.

[12] C. Lyu, D. Gu, Y. Zeng, and P. Mohapatra, "Pba: Prediction-based authentication for vehicle-to-vehicle communications," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 1, pp. 71–83, 2016.

[13] E. L. Cominetti, M. V. M. d. Silva, M. A. Simplício Jr., H. K. Patil, and J. E. Ricardini, "Faster verification of V2X basic safety messages via message chaining," *Vehicular Communications*, vol. 44, p. 100662, 2023.

[14] H. Jin and P. Papadimitratos, "Proactive certificate validation for VANETs," 2020, bloom Filter-based pseudonym validation.

[15] A. Broder and M. Mitzenmacher, "Network applications of bloom filters: A survey," *Internet Mathematics*, vol. 1, no. 4, pp. 485–509, 2004.

[16] L. Cao, H. Yin, J. Hu, and L. Zhang, "Performance analysis and improvement on DSRC application for V2V communication," in *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*. IEEE, 2020, pp. 1–6.

[17] V. Sharma, I. You, and N. Guizani, "Security of 5G-V2X: Technologies, standardization, and research directions," *IEEE Network*, vol. 34, no. 5, pp. 306–314, 2020.

[18] J. B. Kenney, "Dedicated short-range communications (dsrc) standards in the united states," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162–1182, 2011.

[19] B. Brecht, D. Therriault, A. Weimerskirch, W. Whyte, V. Kumar, T. Hehn, and R. Goudy, "A security credential management system for V2X communications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 12, pp. 3850–3871, 2018.

[20] C. Zhang, R. Lu, X. Lin, P.-H. Ho, and X. Shen, "An efficient identity-based batch verification scheme for vehicular sensor networks," in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, Phoenix, AZ, USA, Apr. 2008, pp. 246–250.

[21] F. Pollicino, D. Stabili, L. Ferretti, and M. Marchetti, "An experimental analysis of ECQV implicit certificates performance in VANETs," in *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*, 2020, pp. 1–6.

[22] A. Perrig, R. Canetti, J. D. Tygar, and D. Song, "Timed efficient stream loss-tolerant authentication (tesla): Multicast source authentication transform," RFC 4082, Jun. 2005. [Online]. Available: https://www.rfc-editor.org/rfc/rfc4082

[23] A. Studer, F. Bai, B. Bellur, and A. Perrig, "Flexible, extensible, and efficient VANET authentication," *Journal of Communications and Networks*, vol. 11, no. 6, pp. 574–588, 2009.

[24] C. Lyu, D. Gu, Y. Zeng, and P. Mohapatra, "Pba: Prediction-based authentication for vehicle-to-vehicle communications," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 1, pp. 71–83, 2016.

[25] M. Khodaei and P. Papadimitratos, "Efficient, scalable, and resilient vehicle-centric certificate revocation list distribution in VANETs," in *Proceedings of the 11th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec '18)*. Stockholm, Sweden: ACM, Jun. 2018, pp. 1–12.

[26] D. Wei, X. Xu, S. Mao, and M. Chen, "Optimizing communication and device clustering for clustered federated learning with differential privacy," *IEEE Transactions on Mobile Computing*, pp. 1–14, 2025, early access.