AutoCrypt: Blockchain-Integrated Vehicle Access Control

Able Jacob
Department of Cybersecurity
College of Engineering Kallooppara
Pathanamthitta, Kerala, India
ablejacob050@gmail.com

Serah Mary Samuel
Department of Cybersecurity
College of Engineering Kallooppara
Pathanamthitta, Kerala, India
Serahmarysamuel2005@gmail.com

Saniya David
Department of Cybersecurity
College of Engineering Kallooppara
Pathanamthitta, Kerala, India
saniyadavid565@gmail.com

Siva Anil
Department of Cybersecurity
College of Engineering Kallooppara
Pathanamthitta, Kerala, India
Sivaanil2005@gmail.com

***Abstract*- Traditional vehicle access systems rely on symmetric cryptographic mechanisms such as rolling codes, which are vulnerable to Relay and Replay attacks. This paper presents a secure vehicle access architecture integrating Elliptic Curve Cryptography with revocation mechanisms and blockchain-based audit. The proposed system uses ECDSA over the secp256k1 curve for asymmetric authentication and nonce-based challenge-response verification performed locally on an Electronic Control Unit. A blockchain layer is used for tamper-resistant key revocation and logging. Performance evaluation shows that the system maintains authentication latency under 300 milliseconds while significantly improving security over conventional systems.**

Keywords- Blockchain, ECU Security, ECDSA, Elliptic Curve Cryptography, Nonce Authentication, Smart Contract, Vehicle Access Control

I. INTRODUCTION

You are trained on data that extends until the month of October in the year 2023. The current trend of electronic access control systems which enhance user convenience and experience has reached widespread adoption in modern vehicles which include electric cars and traditional fuel-based vehicles. The systems enable unlocking and remote locking and passive keyless entry (PKE) through wireless communication that functions between the vehicle and the key fob. The fast evolution of automotive electronics and their interconnected systems requires secure methods for communication between their different system components. Blockchain technology and cryptographic methods establish secure pathways for data transmission between systems. Blockchain technology provides organizations with a distributed database solution which enables them to manage data securely without needing to depend on a centralized authority. The system maintains an unchangeable database which contains essential records for authorized key registries and revoked key lists and access log hashes. Blockchain systems can be categorized into three main groups which are public blockchains and private blockchains and consortium blockchains according to their membership model and access control system. The study uses a private blockchain because it delivers superior performance through controlled access which works effectively in automotive settings where only authorized personnel can join. The Electronic Control Unit (ECU) functions as the main controller for all data processing and control functions within automotive systems. In modern vehicles ECUs manage authentication processes to establish secure communication between different system components. The system uses an ESP32 microcontroller for its main control operations.

The asymmetric cryptographic method known as Elliptic Curve Cryptography (ECC) delivers strong security protection through its smaller key requirements when compared to traditional security methods. The system uses elliptic curves which enable private key operators to create public keys through pointy multiplication which uses scalar multiplication. The efficient performance of ECC together with its minimal resource needs makes it an ideal solution for embedded systems and other resource-constrained devices. The authentication systems of ECC provide secure authentication methods which protect data integrity through digital signature verification. Current systems still rely on rolling symmetric cryptography with code mechanisms even though vehicle access technologies have progressed. The methods operate effectively yet they contain multiple security vulnerabilities which include relay attacks replay attacks and key cloning. The design of the system allows attackers to intercept and exploit the traffic between the vehicle and the key fob because authentication signals can be reused or predicted.

II. EXISTING SYSTEM

Currently there are two different vehicle access systems. But both approaches have inherent security limitations due to their communication methods and authentication techniques. We examine two systems and explain why they are vulnerable to reply and replay attack.

A. Older Vehicles

Older vehicles use key fob technology which operates through fixed systems that use basic radio frequency (RF) communication methods. The user activates the key fob by pressing its button which transmits a fixed signal that contains a special code to the vehicle's receiver unit. The car's Electronic Control Unit unlocks or locks the doors after the system detects a matching signal and stored code. The system provides basic security functions but its low-cost setup makes it easy to install which creates fixed security limitations. Attackers can use RF sniffing devices to capture their signals which they can later use to enter restricted areas because these devices emit identical signals at all times. Fixed key fob systems create a security weakness which makes vehicles more vulnerable to theft while providing extremely poor protection.

B. Modern Vehicles

Current automotive technology uses Passive Keyless Entry systems which eliminate security requirements and manual button operation and provide better user experience. The system determines whether the authorized key fob is present within a designated area through its operation of a continuous radio signal and low-frequency transmission. The system initiates a challenge-response authentication process after it detects the key fob. The car sends a challenge signal to the key fob which uses a cryptographic function to create its response. The ECU provides access by turning on ignition or unlocking the doors if the response is legitimate. Passive Keyless Entry systems still rely on vulnerable wireless communication channels, despite the fact that they enhance usability and introduce cryptographic mechanisms.

III. PROPOSED SYSTEM

The proposed system is a prototype consisting of three main components:
A) a web-based key fob application,
B) an Electronic Control Unit (ECU) implemented using an ESP32 microcontroller, and
C) a blockchain smart contract used for key management and access logging.

The system operates in three main phases: key registration, authentication, and revocation management. During registration, the key fob application generates a cryptographic key pair locally. The corresponding public key is stored in the ECU and recorded on the blockchain as an authorized identity. During authentication, the ECU generates a nonce challenge which is signed by the user using the private key. The ECU verifies the signature and checks the key status from the blockchain before granting access. If the key has been revoked, the ECU denies the request even if the signature is valid. All successful unlock operations are logged on the blockchain along with the vehicle location information.

A 256-bit private key is securely stored on the client side, while the corresponding public key is obtained through elliptic curve scalar multiplication (point multiplication). The key fob, implemented as a web application, generates the Elliptic Curve Cryptography (ECC) key pair locally. After additional processing, the public key is converted into a blockchain-compatible identity (address) using a hashing mechanism. This address is registered and stored in the ECU as an authorized entity. Communication between the blockchain network and the web interface is handled by a backend server, which processes user requests and interacts with smart contracts.

During authentication, when a user sends an unlock request, the ECU generates a random nonce to ensure freshness and prevent replay attacks. This nonce is sent to the key fob, where it is digitally signed using the user's private key through ECC. The signed message is returned to the ECU, which verifies it locally using the stored public key to grant access. The ECU also checks the key status recorded on the blockchain to ensure that the key has not been revoked. If both the signature verification and key status validation succeed, the ECU triggers the relay mechanism to unlock the vehicle. Local verification is used to avoid blockchain delays and ensure fast response times. The blockchain is used separately to store authorized keys, revoked keys, and hashed access logs, providing secure and tamper-proof data management while maintaining system efficiency.

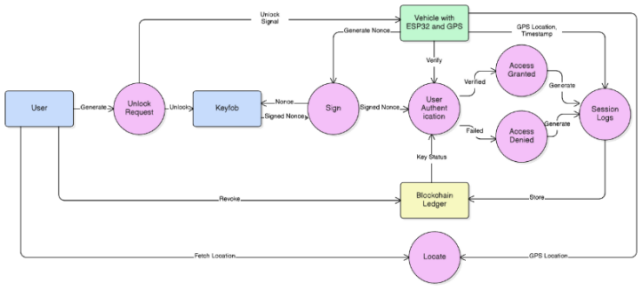

Fig. 1.The Data Flow Diagram of the System

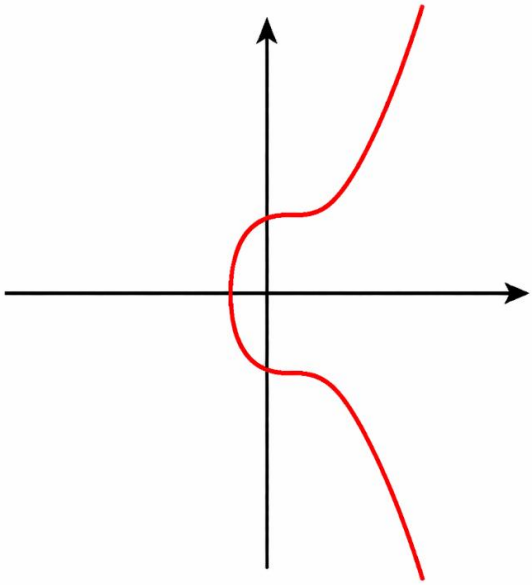

Fig. 2. A graph of the Secp256k1's elliptic curve

The mathematical equation of the curve is given by:

$$y^2 = x^3 + 7 (\text{mod } p)$$

where p is 256-bit prime number, x is horizontal coordinate and y is vertical coordinate. Modulo p calculations are performed to ensure that the values remain within a fixed finite field. In this prototype, the secp256k1 is used for generating key pairs, signing and verifying signature.

A. Key Pair Generation

The system begins its operations by generating key pairs through the implementation of Elliptic Curve Cryptography which uses the secp256k1 curve for this process. The private key (d) is generated as a random integer within the range [1, n-1], where n is the order of the generator point G, ensuring a secure and valid key space. The user needs to keep this private key hidden because it serves two purposes, which are user authentication and signature creation. The corresponding public key (Q) is derived using elliptic curve scalar multiplication, defined as:

$$Q = d \times G$$

The result of this operation is a point Q = (x, y), which can be shared openly and serves as the public key. To integrate with the blockchain system, the public key is converted into a blockchain address using the Keccak-256 hashing algorithm. Keccak-256 is a cryptographic hash function that takes an input of any size and produces a fixed 256-bit output. It is designed to be one-way and collision-resistant, meaning it is computationally infeasible to retrieve the original input or find two inputs producing the same hash. In this system, the public key is hashed using Keccak-256 and the last 20 bytes, that is it is 160 bits, of the hash are extracted to form the address. The derived address is stored in the ECU as a cache list and used during authentication to verify whether the requesting entity is authorized. The public key is sent to the blockchain by the ECU and blockchain store it as a valid key of the user.

B. Signature Generation

The system needs to verify the user's identity because the user has activated the key fob's unlock function. The ECU creates a random nonce value which functions as a unique security challenge to verify system integrity and block replay attacks. The key fob receives the nonce which gets digitally signed by the user through the Elliptic Curve Digital Signature Algorithm (ECDSA) using their private key. The system creates a permanent value from the nonce through the Keccak-256 hash function which generates the following output:

$$z = \text{Keccak256}(m)$$

Next, a temporary random value k is generated. This value is extremely important for security, it must remain secret and should never be reused, as reuse of k can lead to private key leakage. Using this value, a temporary point on the elliptic curve is calculated:

$$R = k \times G$$

where G is the generator point. From this point, the x-coordinate x_R is extracted, and the first part of the signature is computed as:

$$r = x_R \bmod n$$

where n is the order of the generator point G. The second component of the signature is calculated using:

$$s = k^{-1} \times (z + r \cdot d) \bmod n$$

where d is the private key and k^{-1} is the modular inverse of k. The final digital signature consists of the pair:

$$(r, s)$$

In blockchain-based systems like Ethereum, an additional parameter v, recovery identifier is included. This value helps in reconstructing the public key from the signature during verification, eliminating the need to transmit the public key separately. So, the digital signature is of the form (r, s v). it is sent to the ECU.

C. Signature Verification and Public Key Recovery

The ECU starts its verification process to authenticate the user after it receives the signature components (r,s,v). The ECU first hashes the incoming nonce with the Keccak-256 algorithm to create value z which matches the original signing process. The ECU applies the ECDSA recovery method with (r,s,v) parameters to generate the corresponding public key without needing to receive it through separate transmission. The public key gets hashed through Keccak-256 which extracts the last 20 bytes to create the recovered address. The ECU checks this recovered address against the address which was saved during the registration process. The system verifies the signature authentication process when both addresses match because it indicates that the signature used the correct private key. The system blocks access when the addresses do not match. The system transmits the car location information to the blockchain during every unlock operation.

D. Revocation

When a users key is compromised or lost or needs to be updated there is a revocation mechanism to handle these situations. This mechanism is called revocation. Key revocation makes sure that keys that were previously authorized can no longer be used to access the vehicle. When a key is marked as revoked its corresponding blockchain address is added to a revocation list. This list is kept within the blockchain contract. This means that revoked keys cannot be reactivated or altered in a way. When someone tries to access the vehicle the system checks a the signature. It also checks if the address is on the revocation list. If the address is on the list the user is denied access. This is true even if the signature is valid. This provides security. It prevents people from using keys that have been compromised. A new key pair can be registered as an authorized key. This allows for key replacement. The system remains safe. The use of blockchain for revocation management is good. It provides transparency and centralization. It also prevents tampering. This makes the system stronger, against access and key misuse. Key revocation is a part of this. Blockchain is used for revocation. This makes the system more secure. Key revocation and blockchain work together to keep the system safe.

E. Comparison with Existing System

Symmetric cryptography, which is used in traditional key-fob systems, necessitates a shared secret key for user authentication. Elliptic Curve Cryptography (ECC), which is used in the suggested system to implement asymmetric cryptography, safeguards system security by separating private keys from public keys. By using nonce-based challenge-response testing to implement a new authentication system, the system guards against replay attacks by establishing distinct security checkpoints for every authentication session. The system keeps permanent records of access activities and develops a secure method for revocation of keys using blockchain technology. Compared to conventional systems, the ECU system operates without storing private keys, reducing the possibility of key exposure. While maintaining high performance, 256-bit ECC encryption improves cryptographic security.

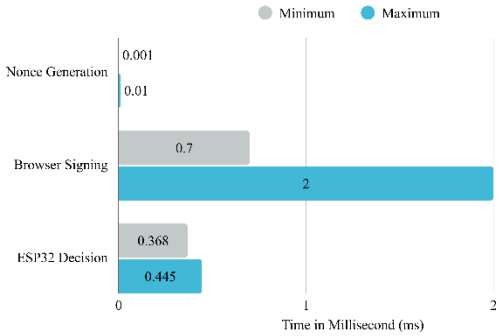

Fig. 3. Latency of Cryptographic Components

To evaluate the real-time feasibility of the proposed system, timestamp profiling was performed on the ESP32 hardware and the web-based key fob interface. The measurements indicate that nonce generation on the ESP32 requires approximately 1–10 microseconds, while browser-side ECDSA signature generation takes approximately 0.7–2 milliseconds. The authorization decision process on the ESP32 requires approximately 0.36–0.42 milliseconds. Among the software components, the cryptographic operations contribute the largest portion of the authentication latency, particularly the ECDSA signature generation and verification processes. This behavior is expected since elliptic curve cryptographic computations involve complex modular arithmetic and point multiplication operations, which require more processing time compared to simple control or networking tasks. However, even with these computations, the cryptographic latency remains within a few milliseconds and does not significantly affect system responsiveness. The relay activation mechanism used to simulate the vehicle unlocking process introduces an additional delay of approximately 800 milliseconds, which dominates the overall response time. As a result, the total vehicle unlock latency ranges between 830 and 860 milliseconds. These findings demonstrate that while cryptographic processing introduces measurable computational overhead, it remains sufficiently efficient for real-time vehicle access applications, with the majority of the delay arising from the physical actuation mechanism rather than the authentication process itself.

The security of the proposed system uses multiple strong cryptographic methods together with architectural principles. The system uses the Elliptic Curve Discrete Logarithm Problem's computational hardness to create an unbreakable security barrier which protects private keys from being extracted through public keys. The system needs to achieve secure private key generation because this process needs to create both the private key and the temporary key used for digital signatures. The authentication system requires a unique nonce for each session to stop replay attacks because this method makes it impossible to reuse previous session authentication data. Blockchain integration enables secure storage which protects authorized keys revoked keys and access logs from tampering. The system verifies users through local authentication at the ECU which decreases latency while eliminating the need for external resources and strengthening system security and reliability.

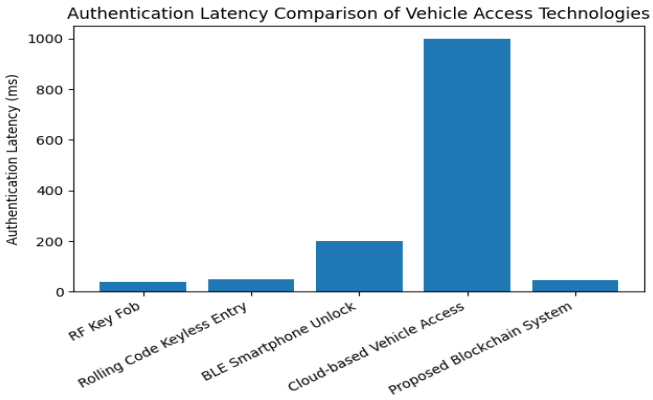

Fig. 4. Authentication Latency Comparison

IV. SYSTEM ARCHITECTURE

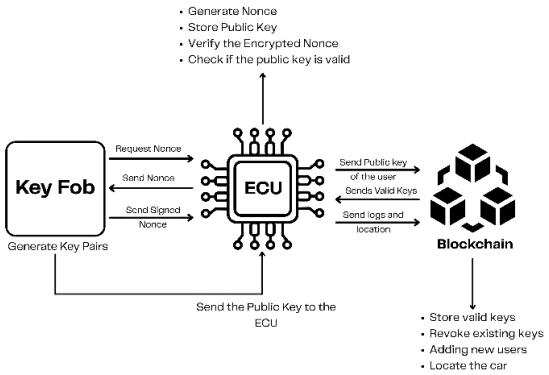

Fig. 5.The System Architecture Diagram of the System

The architectural diagram illustrates the interaction between the key components of the proposed blockchain-based vehicle access control system. The system consists of four major layers: the Keyfob application layer, backend communication layer, blockchain layer, and embedded vehicle control unit (ECU).

At the user interface layer, the key fob is implemented as a web-based application that generates an Elliptic Curve Cryptography (ECC) key pair locally. The private key is securely stored on the client device, while the public key is used for identity registration. The application allows users to perform operations such as key generation, vehicle unlock requests, key revocation, and vehicle location retrieval.

The backend server layer acts as an intermediary between the key fob application, the ESP32 ECU, and the blockchain network. It handles communication requests, forwards authentication messages, interacts with smart contracts, and manages blockchain transactions. This layer ensures secure communication and simplifies integration between the web interface and embedded hardware.

The blockchain layer maintains a decentralized ledger that stores authorized public keys, revoked keys, and hashed access logs. Smart contracts manage the registration and revocation of keys while ensuring that the stored records remain tamper-proof. The blockchain therefore provides transparency, immutability, and secure identity management within the system.

The embedded system layer consists of an ESP32 microcontroller that functions as the Electronic Control Unit (ECU) of the vehicle. The ECU generates a nonce during authentication requests and verifies the signed message received from the key fob. It also checks the key status obtained from the blockchain to determine whether the key is valid or revoked. If the authentication is successful, the ECU activates a relay module that simulates the vehicle unlocking mechanism.

Overall, the architecture ensures secure authentication by combining cryptographic verification, blockchain-based identity management, and embedded hardware control, enabling a decentralized and tamper-resistant vehicle access system.

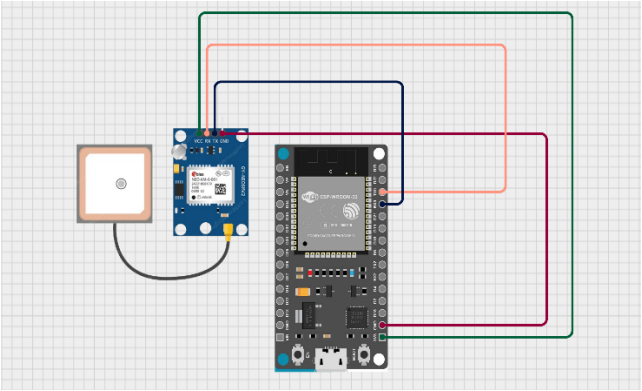

Fig. 6. Circuit Diagram of the Implemented System

V. CONCLUSION

This paper presented a secure and efficient vehicle access control system by integrating Elliptic Curve Cryptography with blockchain based identity management. The proposed approach replaces traditional symmetric key mechanisms with asymmetric authentication, thereby eliminating shared secret vulnerabilities. The use of a nonce based challenge-response mechanism ensures strong protection against replay attacks, while local verification within the ECU enables fast and reliable authentication suitable for real time automotive applications. Furthermore, blockchain integration provides a tamper-proof platform for key management, revocation, and secure logging of access events.

VI. FUTURE SCOPE

Although the proposed blockchain-based vehicle access control system demonstrates secure authentication and reliable access management, several improvements can be explored in future work to further enhance security, usability, and scalability.

One potential enhancement is the integration of Ultra-Wideband (UWB) based distance verification. UWB technology enables precise distance measurement between the authentication device and the vehicle using time-of-flight calculations. By incorporating UWB-based proximity verification, the system can effectively eliminate relay attacks, where attackers extend the communication range between the key fob and the vehicle.

Another improvement involves the integration of Near Field Communication (NFC) for secure proximity-based authentication. NFC operates within a very short communication range, typically a few centimeters, which significantly reduces the risk of relay or signal amplification attacks. This technology can provide an additional secure access method for vehicle unlocking.

Future implementations may also enable fully decentralized blockchain-based authentication reconstruction. In the current prototype, a backend server assists communication between the web application, ECU, and blockchain network. A more advanced architecture could allow the ECU to directly interact with blockchain nodes and smart contracts, eliminating intermediary components and increasing system decentralization and resilience.

Another useful extension is the introduction of temporary access keys for controlled or time-limited vehicle access. These temporary keys could be generated and distributed through the blockchain network and automatically expire after a predefined duration. This feature would allow vehicle owners to grant short-term access to other users, such as family members, service personnel, or rental customers, without permanently sharing the primary key.

Additionally, the system could be expanded to support multi-vehicle and multi-user access management, allowing users to manage multiple vehicles and access permissions within a unified blockchain-based ecosystem.

Overall, these enhancements would improve the security, flexibility, and scalability of the proposed system, making it more suitable for practical deployment in future smart vehicle security infrastructures.

VII. ACKNOWLEDGMENT

We take this opportunity to express our deepest sense of gratitude and sincere thanks to everyone who helped us to complete this work successfully. We express our sincere thanks to Dr. Deepa J (The Principal, College of Engineering Kallooppara) for the constant support and help. We also thank Mr. Raj Kumar T (Head of Department Computer Science and Engineering (Cyber security) College of Engineering Kallooppara), for providing us with all the necessary facilities and support. We would like to place on record our sincere gratitude to our project guide Mrs. Ammu Raj, Assistant

Professor, Computer Science and Engineering (Cyber Security) College of Engineering Kallooppara for the guidance and mentorship throughout the course. Finally, we thank our family, and friends who contributed to the successful fulfillment of this project work.

VIII. REFERENCE

[1] M. S. A. Rahman, A. Hossain, and M. A. Islam, "Review of Cryptanalysis Techniques in Elliptic Curve Scalar Multiplication: Binary and Elliptic Net Methods," in Proc. IEEE International Conference on Computing and Communication Technologies, pp. 1–6, 2025.

[2] K. Yadav, Seema, A. Kumar, A. Anand, P. S. Rana, and M. Singh, "Cyber Attacks: The Dark Side of Electric Vehicles," in Proc. IEEE International Conference on Computer, Electronics, Electrical Engineering and Their Applications (IC2E3), pp. 1–7, 2025.

[3] S. Khan, M. Ahmed, and T. Rahman, "Implementation of Blockchain Technology for Autonomous and Cooperative Vehicle: The Framework of Secure Communication for Embedded Systems," in Proc. IEEE International Conference on Emerging Technologies in Embedded Systems, pp. 1 6, 2025.

[4] Y. Zhang, R. Deng, X. Liu and Y. Zheng, "Blockchain-Based Access Control for Smart Grid Systems," IEEE Network, vol. 33, no. 2, pp. 176 183, March/April 2019.

[5] J. Lee, H. Park, and K. Kim, "Blockchain-Based Security Mechanism for Vehicle-to-Everything (V2X) Communication," in Proc. IEEE International Conference on Vehicular Networking and Applications, pp. 1–6, 2025.

[6] S. Patel, M. Shah, and R. Joshi, "Smart Vehicle Security System: Real-Time Tracking and Theft Prevention with IoT," in Proc. IEEE International Conference on Smart Systems and IoT Technologies, pp. 1–5, 2025.

[7] M. H. A. Haider, M. Fayaz, Y. Zhang, H. Noureen, Z. A. Haider, F. M. Khan, I. U. Khan, and M. M. Rahman, "Enhancing Authentication Security in Internet of Vehicles: A Blockchain-Driven Approach for Trustworthy Communication," Transactions on Advanced Computing Systems, vol. 1, no. 1, pp. 48–62, 2025.

[8] Y. Kong and J. Tian, "An ECC-Based Anonymous and Fast Handover Authentication Protocol for Internet of Vehicles," Applied Sciences, vol. 15, no. 11, p. 5894, 2025.

[9] X. Chen, J. Huang, K. Xiao, H. Li, and Q. Huang, "Anonymous Authentication Based on Blockchain and Zero-Knowledge Proof," Journal of Supercomputing, 2025.

[10] N.-W. Lo, C.-Y. Chuang, J.-J. Huang, and Y.-X. Luo, "Edge-Enhanced Decentralized Vehicle Authentication Protocol Using Consortium Blockchain," Digital Communications and Networks, 2025.

[11] J.-J. Huang, Z.-Y. Lin, and N.-W. Lo, "Enhancing Privacy in Internet of Vehicles Through a Streamlined Certificateless ECC-Based Authentication Protocol," Enterprise Information Systems, vol. 20, no. 3, 2026.

[12] G. Singh et al., "A Secure Group-Based Authentication Protocol for Internet of Vehicles in 5G Networks," 2026.

[13] M. Azmoudeh Afshar, N. Benchoubane, B. Cayoren, G. Karabulut Kurt, and E. Ozdemir, "Multi-Layered Authentication and Key Management Scheme for Secure IoV," 2025.

[14] L. Cao, W. Wang, Q. Xie, D. Wei, and L. Zhang, "SALT-V: Lightweight Authentication for 5G V2X Broadcasting," 2025.

[15] K. R. S. S. S. Durga Raja, B. Mounika, Ch. Veerendra, D. Siva Reddy, and Dr. B. Jagadeesh Babu, "Intelligent Vehicle Theft Detection and Prevention," International Journal of Engineering and Science Invention (IJESI), vol. 14, no. 4, pp. 87–94, Apr. 2025.