



ARTICLE

Light-Weighted Mutual Authentication and Key Agreement in V2N VANET

Yanan Liu¹, Lei Cao^{1,*}, Zheng Zhang^{1,*}, Ge Li², Shuo Qiu¹ and Suhao Wang¹

¹School of Network Security, Jinling Institute of Technology, Nanjing, 211169, China

²School of Command and Control Engineering, Army Engineering University of PLA, Nanjing, 210007, China

*Corresponding Authors: Lei Cao. Email: 2307050001@stu.jit.edu.cn; Zheng Zhang. Email: zhangzheng@jit.edu.cn

Received: 18 April 2025; Accepted: 05 June 2025; Published: 30 July 2025

ABSTRACT: As the adoption of Vehicular *Ad-hoc* Networks (VANETs) grows, ensuring secure communication between smart vehicles and remote application servers (APPs) has become a critical challenge. While existing solutions focus on various aspects of security, gaps remain in addressing both high security requirements and the resource-constrained nature of VANET environments. This paper proposes an extended-Kerberos protocol that integrates Physical Unclonable Function (PUF) for authentication and key agreement, offering a comprehensive solution to the security challenges in VANETs. The protocol facilitates mutual authentication and secure key agreement between vehicles and APPs, ensuring the confidentiality and integrity of vehicle-to-network (V2N) communications and preventing malicious data injection. Notably, by replacing traditional Kerberos password authentication with Challenge-Response Pairs (CRPs) generated by PUF, the protocol significantly reduces the risk of key leakage. The inherent properties of PUF—such as unclonability and unpredictability—make it an ideal defense against physical attacks, including intrusion, semi-intrusion, and side-channel attacks. The results of this study demonstrate that this approach not only enhances security but also optimizes communication efficiency, reduces latency, and improves overall user experience. The analysis proves that our protocol achieves at least 86% improvement in computational efficiency compared to some existed protocols. This is particularly crucial in resource-constrained VANET environments, where it enables efficient data transmission between vehicles and applications, reduces latency, and enhances the overall user experience.

KEYWORDS: Kerberos; PUF; authentication; key agreement; VANET

1 Introduction

The remarkable progress of modern communication and Internet technology has propelled the Vehicular *ad-hoc* network (VANET) technology to new heights. VANET represents a revolutionary convergence of multiple technological domains, integrating an array of components including sensors, On-Board Units (OBUs), and other advanced devices [1]. This integration serves as the cornerstone for realizing intelligent information interaction across the vehicle-to-everything (V2X) framework. Through V2X, vehicles can communicate with other vehicles (V2V), roadside infrastructure (V2I), networks (V2N), and even pedestrians (V2P), creating a seamless and highly interconnected transportation ecosystem.

When considering the application scenarios of VANET, two main categories emerge based on distinct interactive objects. The first scenario focuses on constructing a travel information network and an intelligent transportation system via local connections. This approach heavily depends on short-range communication technologies like Dedicated Short Range Communication (DSRC) or cellular-based communication technologies such as Cellular-Vehicle-to-Everything (C-V2X). DSRC, for example, enables vehicles to exchange information within a relatively short distance, typically up to a few hundred meters. It can be used for



applications like intersection collision avoidance, where vehicles can transmit their speed, direction, and position to nearby vehicles and infrastructure, helping to prevent accidents [2]. C-V2X, on the other hand, leverages cellular networks, offering a broader communication range and higher data transfer rates [3]. It can support more complex applications, such as real-time traffic management and remote vehicle monitoring. However, despite their potential, these technologies are still in the process of maturation. Issues like signal interference, limited coverage in certain areas, and high-cost deployment remain significant challenges that need to be addressed.

In contrast, Vehicle-to-Network (V2N) technologies, which are closely associated with lightweight VANET applications, are experiencing a period of rapid expansion [4]. V2N technologies open the door to a wide range of applications that significantly enhance the driving experience. Cockpit ecosystems, for instance, have transformed the in-car environment. They now offer personalized dashboards, real-time traffic-based navigation, and integrated entertainment systems. Drivers can customize the display according to their preferences, and the system can provide route suggestions based on up-to-the-minute traffic conditions. Remote vehicle control is another area where V2N shines. Owners can use their mobile phones to lock or unlock their vehicles, start the engine, and even pre-set the climate control system, adding a new level of convenience. Remote fault diagnosis is also a crucial application. By continuously monitoring vehicle data, service providers can detect potential problems early and schedule maintenance promptly, reducing the likelihood of unexpected breakdowns. Big data analytics of vehicle operations further optimizes vehicle performance. By analyzing large volumes of data collected from various sensors in the vehicle, manufacturers can identify areas for improvement, such as fuel efficiency and component durability.

A prime example of a V2N-enabled application is over-the-air (OTA) updates. In this process, the vehicle's communication terminal, often denoted as the TBOX, plays a pivotal role. First, the TBOX establishes communication with roadside units and network servers. These roadside units act as intermediaries, relaying data between the vehicle and the broader network. Once connected to the network, the vehicle can access the automaker's OTA cloud server. Here, it can download upgrade packages that may include improvements to the vehicle's software, such as updated maps for the navigation system, enhanced engine management algorithms, or new features for the infotainment system. However, this process is not without its risks. VANET applications require secure and authenticated communication between the vehicle and the application server (APP). During vehicle-cloud communication, the interaction between on-board terminals and the cloud generates a substantial amount of user privacy data. This data includes sensitive information such as driving habits, location history, and vehicle diagnostic data. Unfortunately, this data is vulnerable to various threats, with network sniffing being a significant concern. Intruders may use network sniffing tools to intercept the data transmitted between the vehicle and the cloud. Once they obtain this information, they can analyze it to gain unauthorized administrative access to the vehicle terminal. In many vehicles, the Controller Area Network (CAN) data bus, which is responsible for transmitting data between different vehicle components, operates in plain text. This makes it an easy target for attackers. If an attacker gains access to the CAN data bus, they can manipulate the signals sent to the Engine Control Unit (ECU), potentially causing the vehicle to malfunction or even putting the driver's safety at risk.

Moreover, on-board terminal hardware is also at risk of forgery attacks. In the absence of mutual identity authentication in vehicle-cloud communication, attackers can create forged cloud service interfaces. These fake interfaces can be used to inject malicious software, such as viruses or worms, into the onboard system. Once inside, the malware can disrupt the normal operation of the vehicle's applications, steal sensitive data, or even take control of critical vehicle functions.

Therefore, establishing a trusted VANET identity, identification, authentication, and key management system is of paramount importance. This system must meet three fundamental security requirements. Firstly,

mutual authentication between the vehicle and the cloud server is essential. This ensures that both parties can verify each other's identity, preventing unauthorized access. Secondly, confidential communication between the vehicle and the cloud is necessary to protect the privacy of user data. Encryption techniques can be used to ensure that data transmitted between the two parties remains unreadable to unauthorized individuals. Finally, integrity assurance of data transmission is crucial. This guarantees that the data received by the vehicle or the cloud is the same as the data that was originally sent, without any modifications or corruption. By fulfilling these requirements, the security of VANET applications can be effectively ensured, safeguarding the safety and privacy of vehicle users [5,6].

With the increasing demand for secure access between vehicles and remote application servers in the V2N environment, traditional authentication mechanisms fail to meet the requirements of security and real-time performance due to their vulnerability to physical attacks and reliance on highly complex public key encryption. In this paper, we will propose an extended Kerberos protocol that incorporates the Physical Unclonable Function (PUF). Tailored for V2N environments, this protocol is designed to facilitate secure access for vehicles to application servers. By implementing mutual authentication and key negotiation mechanisms between vehicles and remote application servers, it ensures the secure and efficient communication that is essential for VANET applications. This not only safeguards the confidentiality and integrity of data transmitted but also enhances the overall security posture of the VANET ecosystem, enabling a more reliable and trusted communication framework for various vehicular services.

2 Related Works

2.1 Authentication and Key Management in VANET

In recent years, the security of the VANET has become a focal point of research, and authentication and key management have emerged as crucial aspects in this domain. As a result, researchers have put forward a plethora of solutions to enhance the security infrastructure of VANET.

In 2020, Chen et al. [7] capitalized on the tamper-proof and distributed nature of blockchain technology. Blockchain, with its decentralized ledger and cryptographic hashing mechanisms, offers a high level of security and immutability. By leveraging these characteristics, they proposed a rapid and anonymous identity authentication scheme. In this scheme, the decentralized structure of the blockchain ensures that no single entity has complete control over the authentication process, reducing the risk of a single point of failure. The anonymity feature is achieved through the use of cryptographic techniques that hide the real identities of vehicles, protecting user privacy. In 2021, Altaf and Maity [8] introduced a mixed signature scheme by combining the Public Key Infrastructure (PKI) and certificate-less signature (CLS). PKI is a widely used framework for managing digital certificates and public-private key pairs, while CLS aims to reduce the reliance on traditional certificates. In their proposed scheme, the tasks of generating pseudo-identities and partial private keys are offloaded to the Trusted Registration Authority (TRA) and Roadside Unit (RSU), which reduces the computational and storage burdens on individual vehicles, thus optimizing resource consumption. In 2022, Mukathe et al. [9] proposed a blockchain-based certificateless authentication scheme for VANETs that combines elliptic curve cryptography (ECC) for signature aggregation and batch verification to address the trade-off between security and efficiency, while leveraging blockchain for decentralized storage and demonstrating resilience against adaptive chosen-message attacks through computational and communication cost analysis. In 2023, Li et al. [10] proposed a double-layer blockchain and decentralized identifiers (DID)-assisted authentication mechanism (BDRA) for VANETs, leveraging DID technology to eliminate single points of failure in registration, combining blockchain and reputation feedback for efficient authentication. In 2024, Guo et al. [11] proposed an innovative approach using an early factorial place division algorithm. This algorithm is designed to implement invalid signature tracking. In a VANET environment,

malicious actors may attempt to forge signatures to gain unauthorized access or disrupt the system. Guo's algorithm can detect such invalid signatures in real-time. It also enables the dynamic revocation of malicious vehicles, ensuring that only legitimate vehicles can participate in the network, thereby enhancing the overall security of the VANET.

2.2 Kerberos Protocols

The Kerberos Protocol [12] is a network authentication protocol proposed by MIT, implementing mutual authentication between client and server or server and server. When users interact with the Kerberos system, they first acquire a Ticket Granting Ticket (TGT) from the Key Distribution Center (KDC) of the Authentication Server (AS). The TGT is then used to request access to the Ticket Granting Server (TGS) for a Service Granting Ticket (SGT) [13]. These tickets contain encrypted session keys for secure communication between the client and the APP.

However, the Kerberos Protocol is not without its vulnerabilities. One of the primary concerns is related to its storage security. The protocol's reliance on password-based authentication mechanisms and symmetric encryption systems exposes it to several security risks. In Kerberos, the short-term session keys generated by the client, AS, and TGS are all stored within the KDC. This centralized storage of keys makes the system vulnerable to various attacks. Password guess attacks pose a significant threat. Since passwords are used as a means of authentication, an attacker can attempt to guess passwords through brute-force or dictionary-based methods. If successful, the attacker can gain unauthorized access to the system. Additionally, replay attacks are a concern. In a replay attack, an attacker intercepts and reuses valid authentication messages, potentially bypassing the authentication mechanism. Key leakage is another major risk. Physical attacks, such as intrusion, semi-intrusion, and side-channel attacks, can be exploited by attackers to obtain the keys stored in the KDC. Once the keys are leaked, the attacker can decrypt communication and gain access to sensitive information. Despite these challenges, the Kerberos Protocol has seen continuous evolution in its application and refinement. Over the years, researchers have been actively working on improving the protocol to enhance its security and performance.

Researchers have been improving the protocol to enhance its security and performance. In 2001, Tung et al. [14] proposed the PKINIT protocol, which integrated PKI into Kerberos. By implementing PKINIT, client identity authentication was no longer solely reliant on passwords. Instead, digital certificates were used to authenticate clients, effectively solving the password-guess problem. In 2018, Sutradhar et al. [15] proposed a new scheme for Kerberos public key extension using the threshold password and ECC. The threshold password improved security by requiring multiple parties to collaborate to access the key. ECC, on the other hand, offered significant advantages in terms of computational efficiency compared to the RSA algorithm used in PKINIT. In 2024, Liang et al. [16] proposed Kerberos-centric security extensions that utilized dynamic ticket-key rotation mechanisms. The optimization incorporated algorithmic credential refreshing and cross-domain authentication protocols, significantly enhancing Kerberos' resistance to replay attacks and MITM threats.

2.3 Physical Unclonable Function (PUF)

PUFs are devices that capitalize on the inherent random variations present in internal gate circuits or connecting circuits to generate unique Challenge-Response Pairs (CRPs) [17]. These random variations occur during the manufacturing process due to minute differences in materials, fabrication tolerances, and environmental factors. PUFs have several remarkable characteristics that make them highly valuable in the realm of security. For example, PUFs are relatively easy to manufacture and also possess the property of unclonability and unpredictability. Collectively, these features of PUFs offer a novel and effective solution

to the long-standing problem of security certification. In traditional security systems, passwords, keys, and digital certificates are used for authentication. However, these can be stolen, lost, or forged. PUF-based authentication, on the other hand, provides a more secure alternative. By using the unique CRPs generated by PUFs, systems can authenticate devices or users with a high degree of confidence, reducing the risk of security breaches.

Challenge-response system is constructed with PUF, generating a unique corresponding response value (response, r) for any challenge value (challenge, c). But the challenge value cannot be deduced from the response value. This structural property of PUF can be viewed as the hardware equivalent of one-way functions. PUF makes it easy to generate a response from a challenge but virtually impossible to reverse-engineer the challenge from the response. This makes PUF-based challenge-response systems highly secure, as they can be used to authenticate the identity of devices or entities without revealing sensitive information about the challenges used.

In this paper, function P is used to identify the unidirectional nature of the PUF:

$$P: C \rightarrow R: P(c) = r, c \in C, r \in R \quad (1)$$

where C represents the challenge set, R represents the response set, c then represents a challenge value in the challenge set C , and r represents the response value corresponding to the challenge value c .

The random variations inherent in a physical object can be thought of as its unique “fingerprint”. These differences are so distinct that instruments are unable to replicate them during the manufacturing process. This irreproducibility gives PUFs the remarkable unpredictable properties. They are also unclonable, since it is impossible to create an exact copy of the unique physical characteristics that determine the PUF’s behavior. Moreover, each PUF is unique, making it an ideal component for secure authentication mechanisms [18]. PUFs can also address the key storage issue in authentication and key negotiation processes. Instead of directly storing keys, PUFs use challenge-response pairs. This approach not only solves the problem of key storage but also effectively prevents key leakage [19]. Additionally, PUF-based hardware is fast-operating and cost-efficient, making it highly suitable for lightweight networks where resources are limited.

In recent years, the PUF structure has been widely used in the field of the Internet of Things (IoT). In 2017, Chatterjee et al. [20] proposed the use of PUF to generate the public identity of each device, and used it as the public key of each device for message encryption, effectively resisting passive and active attacks. In 2019, Li et al. [21] proposed a low-cost IoT secure communication scheme based on PUF and a certificate-free public key cryptography system, which can realize device-to-device secure message transmission without storing secret parameters. In 2021, Idriss et al. [22] proposed a lightweight PUF-based authentication protocol utilizing secret pattern recognition to enable mutual authentication and secure message exchange for IoT devices, avoiding cryptographic functions while maintaining resilience against modeling attacks through nonlinear operations and TRNG integration. In the same year, Liang et al. [23] proposed a method for generating real-time authentication information using deep learning, which solved the key storage problem by using PUF circuit structure labels without storing any key information. In 2023, Lai et al. [24] proposed a PUF-based authentication and key distribution scheme for CAN bus networks that addresses ECU spoofing attacks while reducing message latency, leveraging PUF technology to prevent long-term key leakage and achieve lower computational/communication overhead compared to existing group-based solutions. In the same year, the protocol proposed by Tan et al. [25] employed PUF and built an index map to process the PUF output signal, realizing the secure authentication of vehicles and roadside units, and other vehicles, thereby reducing the risk of the protocol being attacked by machine learning. Shin and Kwon [26] introduced an architecture for integrating WSNs and 5G in IoT based on ECC privacy-preserving authentication, authorization, and key agreement scheme for WSNs in 5G-integrated IoT.

2.4 The Main Problems and Analysis of Existing Work

Table 1 compares the different characteristics of existing identity authentication protocols. Although these works have played an important role in ensuring the security of IoT terminals and communications, there are still the following defects and deficiencies.

Table 1: Summary and comparative analysis of research results on secure identity authentication protocol

Document	Safe objective	Optimize objective	Core technology	Advantages of the programme	Security threats
Chen et al. [7]	Privacy protection	Reduce Authentication delay	Blockchain + Smart Contract	Supports both user partial anonymity and traceability	Physical Attacks
Altaf et al. [8]	Privacy protection	Improve the efficiency of certification in high-density environments	CLS + ElGamal	Uses reputation values and regional vehicles for privacy protection VANET system	Replay Attacks
Mukathe et al. [9]	Identity authentication	Reduce communication overhead	Blockchain	Simplify the certification process and reduce the system burden	Physical Attacks
Li et al. [10]	Identity authentication	Reduce energy consumption	Blockchain	Implement decentralized and efficient user authentication	Physical Attacks
Guo et al. [11]	Privacy protection	Reduce energy consumption	Cryptography	Strong privacy protection features	Physical Attacks
Chatterjee et al. [20]	Identity authentication	Reduce energy consumption	PUF + ECC	No need for public key infrastructure (PKI)	Physical cloning attacks
Li et al. [21]	Confidentiality and identity authentication	Reduce computing overhead	PUF + Non-Intrusive ECC	Support a lightweight message authentication mechanism	Replay Attacks
Lai et al. [24]	Data confidentiality and integrity	Balance security and load	PUF	Reduce the risk of long-term key leakage	Replay Attacks
Tan et al. [25]	Identity authentication	Reduce energy consumption	MI-PUF	Effectively resist machine learning attacks	DoS Attacks
Shin and Kwon [26]	Privacy protection	Improve safety	ECC	Improved user privacy and security	man-in-the-middle attack

Despite the significant amount of foundational research in this area, the proposed security authentication solutions encounter a multitude of challenges when applied in real-world environments. For example, centralized classic PKI certificate management, while capable of achieving anonymous authentication, faces a major drawback as the number of vehicles in the VANET network grows. The centralized nature of PKI means that a single authority is responsible for managing all certificates. As the number of vehicles increases, the processing load on this central authority escalates, leading to significant processing delays. This can be a critical issue, especially in time-sensitive applications such as real-time traffic management. Fast identity authentication protocols based on digital signatures have been designed to enable vehicles equipped with OBUs to quickly gain access through Roadside Units (RSUs). However, in a densely populated area with a large number of OBUs, the RSU's authentication accuracy can be severely compromised. When too many vehicles attempt to authenticate simultaneously, the RSU may experience overload, leading to incorrect authentication decisions. This can potentially result in system failures, as unauthorized vehicles may be granted access, or legitimate vehicles may be denied entry. Blockchain-based fast anonymous identity authentication schemes have shown great promise in leveraging blockchain's tamper-proof and distributed features for efficient security authentication. However, their security is highly dependent on the vehicle identity information stored within the same blockchain. If the blockchain is compromised, for example, through a 51% attack (a situation where an attacker controls more than half of the network's computing power), the entire authentication system can be at risk. In addition, PUF-based identity authentication protocols, while offering hardware-based security by utilizing PUFs, face challenges related to the scalability and reliability of the PUFs in highly dynamic environments like VANETs. PUFs are vulnerable to environmental factors such as temperature and voltage fluctuations, which can affect their stability, leading to authentication failures. Moreover, these schemes mainly focus on device authentication between V2V or V2I, but not V2N.

This paper proposes an improved PUF-based lightweight Kerberos authentication protocol to address challenges in V2N scenarios. By integrating PUF with the Kerberos framework, the protocol achieves efficient mutual authentication and key negotiation while maintaining resistance to physical attacks. Unlike existing PUF-based solutions limited to V2V/V2I contexts, protocol design specifically optimizes for V2N's high-latency tolerance and centralized server architecture, enabling scalable authentication without compromising real-time performance. A novel contribution of this protocol is the incorporation of an environmental monitoring mechanism within the vehicle's PUF module. This system continuously tracks physical variables such as temperature and humidity during challenge-response operations. When significant environmental fluctuations are detected (e.g., deviations $>5\%$ from calibration baselines), the protocol automatically initiates a self-calibration procedure to stabilize the PUF's operating conditions. For example, by activating thermal regulators or humidity controls embedded in the OBU hardware, the module compensates for environmental disturbances before they impact the CRP reliability. These innovations collectively ensure the security and efficiency of identity authentication in complex vehicular network environments and offer a new solution for vehicular network authentication.

3 PuKE-V2N Protocol

This section will introduce the details of a PUF-based Kerberos extension protocol for V2N VANET, which is denoted as the PuKE-V2N protocol.

3.1 System Model

The PuKE-V2N protocol aims to meet the security and efficiency requirements in VANETs. This protocol aims to provide a powerful solution for mutual authentication, key establishment, and secure data

transmission between vehicles and APPs in source-constrained environments, while minimizing computational overhead and latency in VANET systems. As shown in Fig. 1, the V2N VANET communication system presented in this paper comprises five key entities: the Key Distribution Center (KDC), the Authentication Server (AS), the Ticket-Granting Server (TGS), the Application Server (APP) equipped with a PUF module, and the vehicle equipped with an On-Board Unit (OBU) and PUF module.

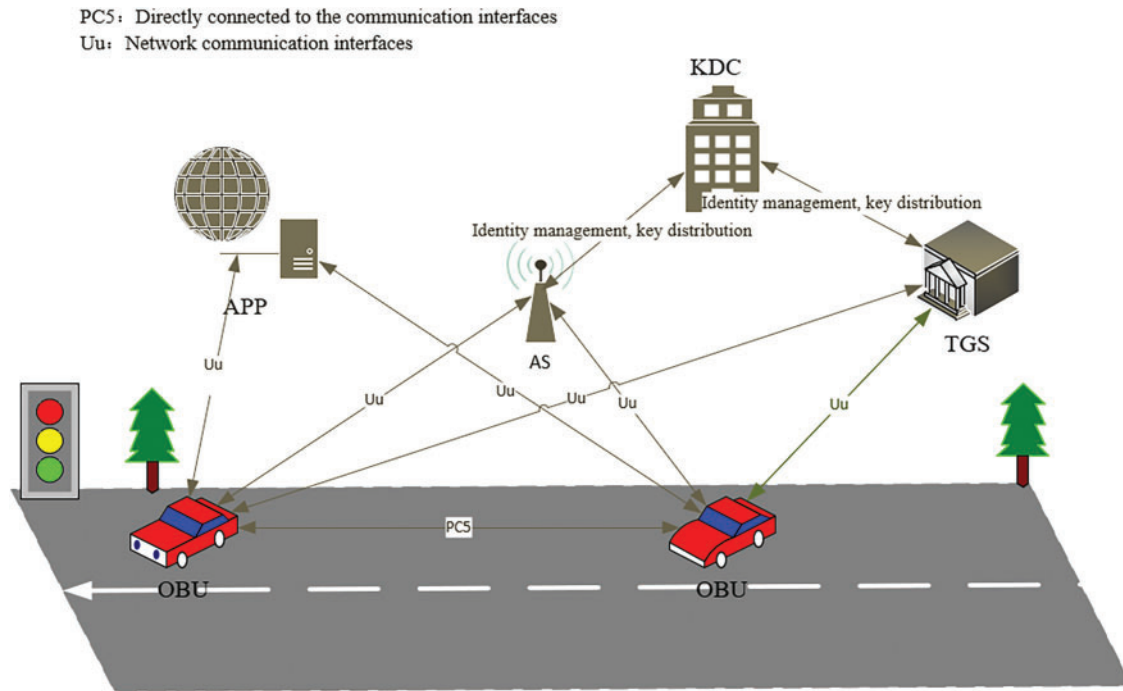


Figure 1: Model diagram of the internet of vehicles communication system

This paper primarily focuses on the identity authentication and key negotiation process between the vehicle and the cloud through the Uu interface, excluding communication between vehicles or with roadside infrastructure units via the PC5 interface. The KDC acts as a trusted third party, providing ticket generation and management services for the entire security authentication process, and is responsible for inspecting and maintaining the PUF module of the vehicles and APPs. The AS within the KDC is responsible for mutual authentication with the client and for generating the TGT for legitimate users. The TGS within the KDC generates the session key for the APP and the service authorization ticket for the client. The OBU in the vehicle is responsible for sending the authentication request to the AS, verifying the AS identity, and establishing the session key with the APP. The APP communicates with the vehicle and establishes the session key in the system.

3.2 Description of the PuKE-V2N Protocol

The PuKE-V2N protocol consists of five stages: registration of vehicle and APP; mutual authentication of vehicle and APP and AS; AS grants TGT to legal vehicles; TGS grants SGT to legal vehicles; establishing session key between vehicle and APP. Table 2 presents the symbols and descriptions used in this article.

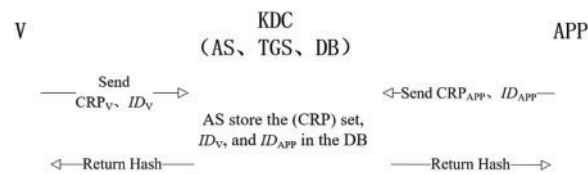
Table 2: Symbol and description

Symbol	Description
V	Vehicle V
APP	Application server
KDC	Key distribution center
AS	Authentication server
DB	Database
TGS	Ticket granting server
TGT	Ticket granting ticket
SGT	Service granting ticket
MAC_V	Message authentication code of vehicle V
CRP_X	PUF challenge-response pairs of device X
$\langle c_V, r_V \rangle$	PUF challenge-response pair of vehicle V
rr_V	The PUF response on vehicle V obtained by inputting a challenge c_V
$K_{X,Y}$	Session key between device X and device Y
ID_X	Identity of the device X
$nonce$	Random number
t	Time-stamp
$Cipher = \{M\}K$	Encrypt plaintext M with the key K to get ciphertext $Cipher$

3.2.1 Registration of V and APP

According to the Kerberos protocol, the KDC consists of the AS, TGS, and a database (DB). Mutual trust is established among these three entities.

Vehicle V is initialized on the AS. First, the vehicle is preset with a PUF structure in a safe environment, recorded as P_V . Generate n random numbers forming the excitation set $\{c_V\}_{i=1,\dots,n}$, and using each excitation input PUF structure to generate the corresponding response gives V the response set $\{r_V\}_{i=1,\dots,n}$. The vehicle then sends the set of all generated excitation response pairs, denoted as $CRP_V = \{\langle c_V, r_V \rangle_i\}_{i=1,\dots,n}$, to the AS secure storage. The APP is initialized on the AS. Preset the PUF structure in a safe environment, denoted as P_{APP} . The APP initializes on the AS as on the vehicle. After initialization, AS stores the set of all excitation response pairs generated by the APP, denoted as $CRP_{APP} = \{\langle c_{APP}, r_{APP} \rangle_i\}_{i=1,\dots,n}$. Incentive response pairs are not stored locally in V and APP, and the details are shown in Fig. 2.

**Figure 2:** V and APP were initialized on the AS

3.2.2 Mutual Authentication of V and APP, and AS

In this stage, the challenge-response mechanism of PUF is used to realize the mutual authentication of the AS and the vehicle V.

Vehicle V sends an authentication request to the AS containing the identity of the vehicle and the TGS. According to the vehicle V identity, AS reads all the incentive response pairs to CRP_V from the database DB and takes one incentive response pair $\langle c_V, r_V \rangle$ to calculate the check code:

$$MAC_V = Hash(r_V) \quad (2)$$

and send the incentive values c_V and MAC_V to the vehicle.

Vehicle V inputs the received c_V into the preset PUF structure P_V to get the corresponding response value rr_V , and then calculates the hash value of rr_V to compare with MAC_V : if equal, the vehicle V authentication AS has succeeded, that is, the vehicle V trusts AS; otherwise, the authentication fails. To prevent direct leakage of CRP_V , the vehicle V calculates the value of $Hash(c_V || rr_V)$ and returns it to AS.

AS calculates the value of $Hash(c_V || rr_V)$ with the received $Hash(c_V || rr_V)$: if equal, the AS-certified vehicle V succeeded, namely AS trusts the vehicle V; otherwise, the authentication fails. As shown in Fig. 3.

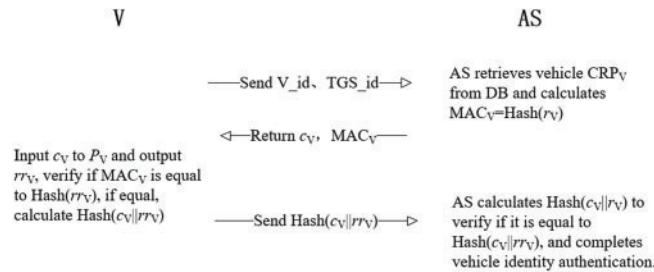


Figure 3: Mutual authentication between AS and V

The APP and AS authentication processes in this scheme are completed before the vehicle V and AS certification. The process is consistent with the vehicle V certification process. After successful authentication, the AS sends the shared key ($K_{TGS,APP}$) between the TGS and the APP to the legitimate APP ($K_{TGS,APP}$ is generated by the PUF response value of the APP).

3.2.3 AS Grants the TGT to Legal V

In this stage, the AS issues a TGT to the authenticated vehicle V.

AS generates session for certified vehicle V with TGS Key ($K_{V,TGS}$). Hashing operations using is XOR values of r_V to generate the vehicle V and AS session key ($K_{V,AS}$), $K_{V,AS}$ encryption $K_{V,TGS}$ to get $\{K_{V,TGS}\}K_{V,AS}$, and then use the master key of TGS (K_{TGS}) encryption $K_{V,TGS}$ to get $\{K_{V,TGS}\}K_{TGS}$. AS generates TGT, TGT includes the identification of vehicle V and TGS, $\{K_{V,TGS}\}K_{TGS}$, t , and *lifetime*. AS sends the TGT and $\{K_{V,TGS}\}K_{V,AS}$ to the vehicle V.

Vehicle V locally hashes the XOR value of rr_V to obtain the session key $K_{V,AS}$ and decrypts the received $\{K_{V,TGS}\}K_{V,AS}$ obtains the vehicle V and TGS session key ($K_{V,TGS}$). As shown in Fig. 4.

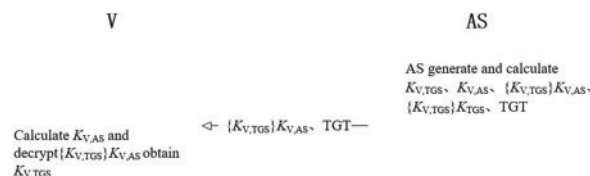


Figure 4: The AS sends the TGT to the vehicle V

3.2.4 TGS Grants the SGT to Legal V

In this stage, the TGS grants the SGT for the legitimate vehicle V.

The vehicle V uses $K_{V,TGS}$ encrypts a random number *nonce* to get $\{nonce\}_{K_{V,TGS}}$ and random number *nonce*, TGT, ID_V , sent to TGS together with ID_{APP} . TGS decryption $\{K_{V,TGS}\}_{K_{TGS}}$ gets $K_{V,TGS}$, and further decrypts $\{nonce\}_{K_{V,TGS}}$ gets *nonce*, compared with the *nonce* received *nonce*: if equal, TGS considers the TGT held by vehicle V is valid as a legitimate vehicle; otherwise, it is invalid.

The TGS generates the session key ($K_{V,APP}$) corresponding APP for legitimate vehicle V, using $K_{V,TGS}$ encryption to obtain $\{K_{V,APP}\}_{K_{V,TGS}}$. Using the shared key of APP and TGS ($K_{TGS,APP}$) to encrypt $K_{V,APP}$ to get $\{K_{V,APP}\}_{K_{TGS,APP}}$. TGS generates SGT, where SGT includes $\{K_{V,APP}\}_{K_{TGS,APP}}$, ID_V , ID_{APP} , t , and *lifetime*, and is sent to vehicle V along with $\{K_{V,APP}\}_{K_{V,TGS}}$, ID_{TGS} . As shown in Fig. 5.

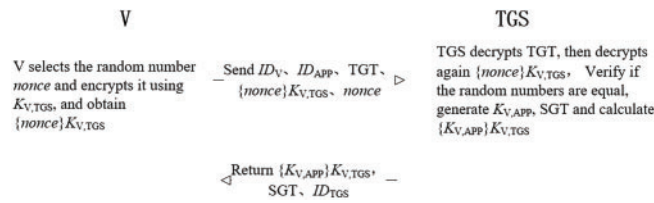


Figure 5: V requests and gets SGT from TGS

3.2.5 V Establishes a Session Key with the APP

Vehicle V uses $K_{V,TGS}$ to decrypt $\{K_{V,APP}\}_{K_{V,TGS}}$ to get $K_{V,APP}$. Vehicle V sends the SGT and ID_V to the APP. After APP receives SGT, APP uses $K_{TGS,APP}$ decrypts SGT to get $K_{V,APP}$ and ID_V . After comparing the received vehicle V identification, the vehicle V and the AS establish a session key and realize confidential communication. As shown in Fig. 6.

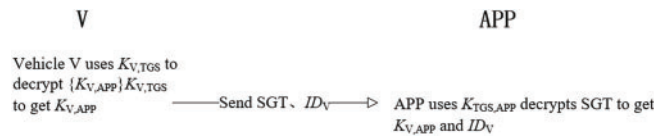


Figure 6: V established secure communication with APP

3.3 PuKE-V2N System Design

In response to the described PuKE-V2N protocol process, this section presents a further implementation plan for the PuKE-V2N system. By combining PUFs with the Kerberos protocol, we utilize the challenge-response mechanism of PUFs to facilitate authentication and key negotiation between vehicles and the APP. This design aims to ensure secure access to cloud servers while the vehicles are in operation, guaranteeing authenticity and confidentiality during data exchanges, and providing resistance against cloning attacks, man-in-the-middle attacks, and replay attacks. The PuKE-V2N System architecture is as shown in Fig. 7.

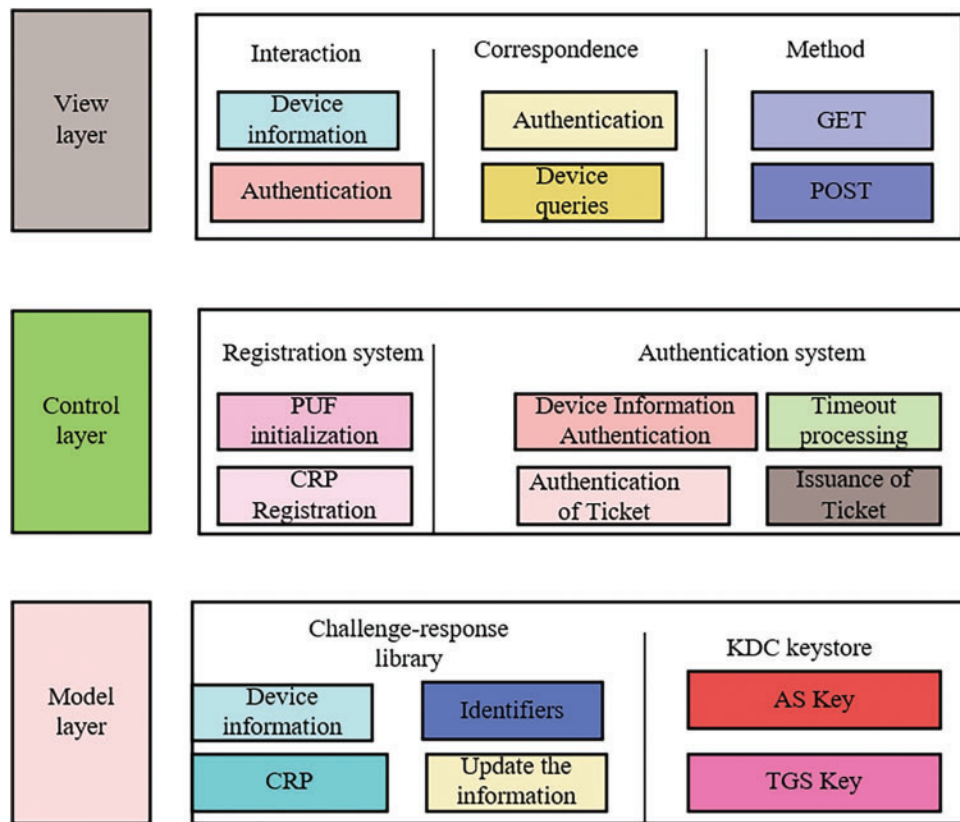


Figure 7: The system is based on the MVC architecture

The interactive process of the system is shown in Fig. 8, including V, APP and KDC. The KDC includes AS, TGS and DB.

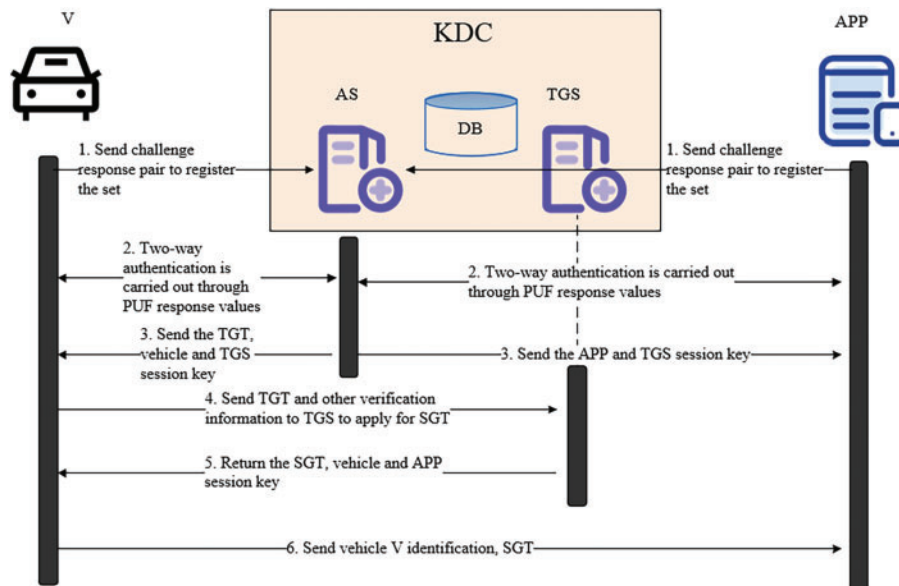


Figure 8: Certification and key negotiation interaction process

4 Security Analysis

The proposed PuKE-V2N scheme, with the support of the KDC, enables mutual authentication and key negotiation between vehicle V and the remote APP. In this scheme, the Challenge-Response Pairs (CRPs) generated by the Physical Unclonable Function (PUF) replace the password-based authentication mechanism of the standard Kerberos protocol. This substitution significantly mitigates the risk of key leakage caused by physical attacks like intrusion, semi-intrusion, or side-channel attacks. By leveraging PUF-generated CRPs, the PuKE-V2N scheme offers a more secure approach to authentication and key management, safeguarding the confidentiality and integrity of communication between vehicles and remote applications in the V2N scenario.

4.1 Formal Analysis

The PuKE-V2N protocol uses the Scyther tool for formal security analysis. Scyther is an open-source formal verification tool dedicated to the automated analysis of security protocols. It is based on model checking technology, exhaustively exploring all possible protocol execution paths, including adversary behaviors, and conforms to the Dolev-Yao network attack model to verify security properties such as confidentiality and authentication. This experiment requires a hardware environment with 8 GB of memory, a Windows 11 operating system, Python 2.7, and the Scyther-w32-v1.1.3 version. In this paper, we define the roles in the protocol as follows: V (Vehicle), AS (Authentication Server), TGS (Ticket Granting Server), and APP (Application Server). Subsequently, the protocol roles and messages must be converted into the input format for Scyther. The Scyther tool uses the SPDL (Security Protocol Description Language) to describe protocols. The SPDL descriptions of the roles in this protocol are shown in Fig. 9.

```

role V {
  fresh vid, tid, aid, sgt, Kvt, Kva, tgt, c, n;
  send_1(V, AS, vid, tid);
  recv_2(AS, V, c, H(puf(c)));
  send_3(V, AS, H(c,puf(c)));
  recv_4(AS, V, tgt, {Kvt}XOR(puf(c)));
  send_5(V, TGS, vid, aid, tgt, n, {n}Kvt);
  recv_6(TGS, V, {Kva}Kvt, sgt, tid);
  send_7(V, APP, vid, sgt);
}

role AS {
  fresh vid, tid, Kvt, tgt, c;
  recv_1(V, AS, vid, tid);
  send_2(AS, V, c, H(puf(c)));
  recv_3(V, AS, H(c,puf(c)));
  send_4(AS, V, tgt, {Kvt}XOR(puf(c)));
}

role TGS {
  fresh vid, tid, aid, sgt, Kvt, tgt, n, Kva;
  recv_5(V, TGS, vid, aid, tgt, n, {n}Kvt);
  send_6(TGS, V, {Kva}Kvt, sgt, tid);
}

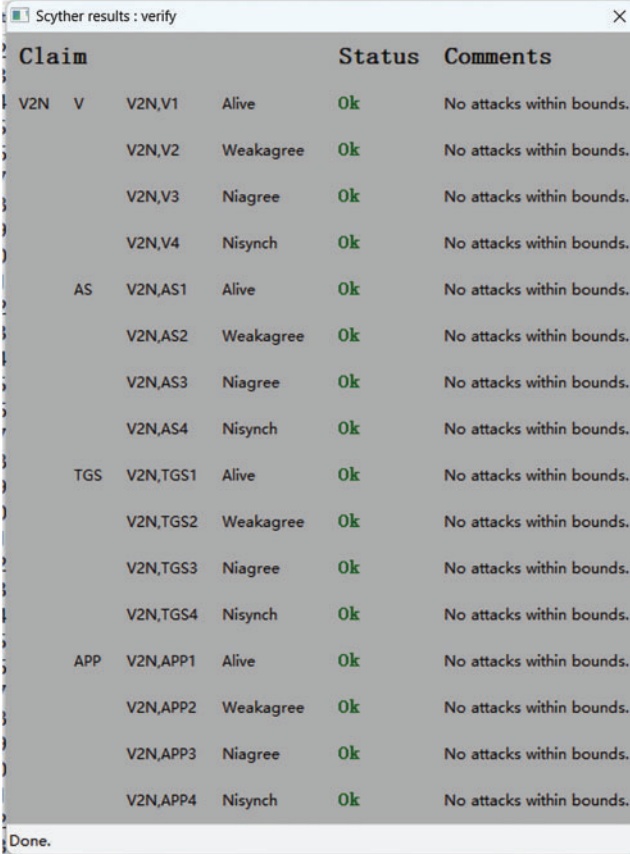
role APP {
  fresh vid, sgt;
  recv_7(V, APP, vid, sgt);
}

```

Figure 9: SPDL language description of each role in this protocol

After the protocol is described, its security is verified. The results of all declarations verified by Scyther are shown in Fig. 10. All declarations in Scyther successfully passed verification, which indicates that even in the event of key leakage, the authentication mechanism can still ensure session security, preserving other security properties and resisting key leakage attacks. In the case of impersonation attacks, by simulating an attacker disguising themselves as a legitimate role, Scyther confirmed that the protocol can identify and reject

the impersonation attack. For man-in-the-middle attacks, the protocol uses encryption and PUF technology to ensure the integrity and confidentiality of messages, preventing the man-in-the-middle from tampering with or decrypting the messages. Additionally, by repeatedly sending legitimate messages, Scyther confirmed that the protocol can prevent attackers from using reflection messages to launch attacks. This indicates that the protocol effectively guarantees confidentiality, authentication, and integrity during the vehicle identity authentication and key agreement processes.



Claim				Status	Comments
V2N	V	V2N,V1	Alive	Ok	No attacks within bounds.
		V2N,V2	Weakagree	Ok	No attacks within bounds.
		V2N,V3	Niagree	Ok	No attacks within bounds.
		V2N,V4	Nisynch	Ok	No attacks within bounds.
	AS	V2N,AS1	Alive	Ok	No attacks within bounds.
		V2N,AS2	Weakagree	Ok	No attacks within bounds.
		V2N,AS3	Niagree	Ok	No attacks within bounds.
		V2N,AS4	Nisynch	Ok	No attacks within bounds.
	TGS	V2N,TGS1	Alive	Ok	No attacks within bounds.
		V2N,TGS2	Weakagree	Ok	No attacks within bounds.
		V2N,TGS3	Niagree	Ok	No attacks within bounds.
		V2N,TGS4	Nisynch	Ok	No attacks within bounds.
	APP	V2N,APP1	Alive	Ok	No attacks within bounds.
		V2N,APP2	Weakagree	Ok	No attacks within bounds.
		V2N,APP3	Niagree	Ok	No attacks within bounds.
		V2N,APP4	Nisynch	Ok	No attacks within bounds.

Done.

Figure 10: Validation results by scyther

4.2 Informal Analysis

Theorem 1: Mutual authentication between vehicle V and AS is achieved in the process of the TGT application.

Proof of Theorem 1: Vehicle V initiates a TGT application to the AS. The AS returns the challenge value c_V and MAC_V to V . V calculates the corresponding response value rr_V from the received c_V using its preset private function P_V , and compares whether $\text{Hash}(rr_V)$ matches MAC_V . Next, V calculates $\text{Hash}(c_V || rr_V)$ and sends it to the AS. The AS compares $\text{Hash}(c_V || rr_V)$ with the received $\text{Hash}(c_V || rr_V)$, confirming that V is a legitimate vehicle. Since only the legitimate AS can compute MAC_V and send it to the vehicle, V verifies the authenticity by comparing the $\text{Hash}(rr_V)$ with MAC_V . Furthermore, the clonal resistance of the PUF ensures that only V can compute the corresponding value $\text{Hash}(c_V || rr_V)$. The AS can also compute $\text{Hash}(c_V || rr_V)$ to verify the vehicle's identity. In conclusion, this scheme achieves mutual authentication between the vehicle and AS during the process of requesting a TGT from the AS. \square

Theorem 2: Mutual authentication between vehicle V and TGS is achieved in the process of the SGT application.

Proof of Theorem 2: The vehicle V initiates the SGT application to the KDC and sends the ciphertext $\{nonce\}_{K_{V,TGS}}$ to TGS, along with the random values $nonce$, TGT , ID_V , and ID_{APP} . After receiving the application, TGS decrypts $\{nonce\}_{K_{V,TGS}}$ in the TGT to obtain the session key $K_{V,TGS}$, and compares the decrypted value of $\{nonce\}_{K_{V,TGS}}$ with the received $nonce$. If they match, TGS considers the TGT held by V as valid and confirms the legitimacy of the vehicle. TGS then generates the SGT for V and the session key $K_{V,APP}$ for the APP, encrypting $K_{V,APP}$ with $K_{V,TGS}$ to obtain $\{K_{V,APP}\}_{K_{V,TGS}}$, which it sends back to V along with ID_{TGS} . Since only a legitimate vehicle can decrypt $K_{V,TGS}$ and authenticate the random $nonce$, TGS can confirm the authenticity of V . Similarly, V authenticates TGS by comparing the returned ID_{TGS} with the expected value. Therefore, this scheme ensures mutual authentication between V and TGS during the process of applying for the SGT. \square

Theorem 3: *Mutual authentication between vehicle V and APP is achieved.*

Proof of Theorem 3: The vehicle V sends the SGT and ID_V to the APP to request access. The APP uses the session key $K_{TGS,APP}$ to decrypt the SGT and retrieve the information contained in the ticket, including the session key $K_{V,APP}$ for communication between the vehicle and APP. The APP authenticates the vehicle's identity by comparing the information in the SGT with ID_V . Since only the legitimate APP certified by AS can access $K_{TGS,APP}$, and unauthorized APPs cannot decrypt the SGT, which ensures the mutual authentication between V and APP, preventing V from accessing an illegitimate APP. \square

Theorem 4: *Resistance against the replay attacks or man-in-the-middle attacks.*

Proof of Theorem 4: In this protocol, a timestamp t and a parameter $lifetime$ are added to the SGT. The presence of the timestamp ensures the uniqueness of the ticket, while the $lifetime$ parameter limits the ticket's validity period. This prevents attackers from successfully carrying out replay or man-in-the-middle attacks, even if they intercept the ciphertext. \square

Theorem 5: *Resistance against forgery attacks and cloning attacks.*

Proof of Theorem 5: During the authentication process, PUFs generate a unique, device-specific identifier based on their non-replicable physical characteristics. As a result, any attempt to forge a vehicle's identity is thwarted. Since the behavior of PUFs cannot be duplicated by external entities, even if an attacker gains access to the system, they cannot generate matching PUF responses. This makes it nearly impossible to impersonate the identity of a vehicle during the authentication process. Consequently, the protocol can resist both spoofing attacks and cloning attacks. \square

Theorem 6: *Resistance against invasion and semi-invasion attacks.*

Proof of Theorem 6: The PuKE-V2N protocol replaces traditional password authentication with CRPs generated by PUFs, significantly enhancing resistance to invasive and semi-invasive attacks. Since CRPs are generated based on the physical characteristics of PUFs and are not stored locally, attackers cannot directly extract the CRP set even if they physically invade the vehicle or server device. AS dynamically and randomly selects CRPs for authentication, and CRPs become invalid after each use, preventing long-term threats from a single exposure. Moreover, during mutual authentication, hash verification and dynamic session key negotiation (e.g., $K_{V,AS}$, $K_{V,TGS}$) ensure that even if an attacker intercepts partial communication data, they cannot forge or replay valid authentication information. Multi-layer encryption (e.g., the generation of TGT and SGT) further isolates the risk of key leakage, making it difficult for invasive or semi-invasive attacks to obtain the complete key chain or tamper with the authentication process. \square

Theorem 7: *Resistance against the side channel attacks.*

Proof of Theorem 7: The protocol inherently resists side channel attacks (SCA) through the physical characteristics of PUFs. The response value r_v generated by PUFs depends on random variations in the chip manufacturing process, which cannot be reverse-engineered to determine the original challenge or response through side-channel techniques such as power analysis or electromagnetic radiation. Additionally, session keys (e.g., $K_{V,AS}$) are derived from dynamically generated PUF responses rather than being statically stored, reducing the window of key exposure. During the authentication process, critical operations (such as hash computation and encryption) only transmit verification values, $\text{Hash}(c_v || r_v)$ instead of the original responses, avoiding the direct transmission of sensitive information over the communication channel and further diminishing the effectiveness of side-channel attacks. \square

Theorem 8: *Resistant against the physical attacks.*

Proof of Theorem 8: The protocol significantly enhances resistance to physical attacks through the physical characteristics of PUF and the dynamic nature of keys. The CRP generated by PUF relies on random physical features during hardware manufacturing (such as chip process variations). Even if an attacker physically intrudes into the device, they cannot directly extract or clone the CRP set. Critical data (such as CRP and session keys) is not stored locally but is generated through secure pre-setting and dynamic negotiation, making it impossible for physical intrusions to obtain long-term valid keys. Additionally, the AS randomly selects CRP for authentication, and it becomes invalid after a single use, preventing long-term risks caused by a single leak. The dynamic response mechanism of PUF and the encryption process still provide a natural barrier against physical attacks, ensuring the security isolation of key generation and transmission. \square

5 Performance Analysis

This protocol is compared with schemes proposed by Li et al. [21], Chatterjee et al. [20], and Shin and Kwon [26]. Chatterjee et al. [20] put forward the utilization of the Physical Unclonable Function (PUF) to generate the public identity for each device. This public identity was then employed as the public key of each device for message encryption purposes. Their approach effectively defends against both passive and active attacks. Li et al. [21] proposed a cost-effective secure communication scheme for the IoT. This scheme was based on the PUF and a certificate-free public key cryptography system. It enabled secure device-to-device message transmission without the need to store secret parameters, thereby simplifying the security infrastructure and enhancing the overall security of IoT devices. Shin and Kwon introduced an architecture for integrating WSNs and 5G in IoT [26]. Based on cryptanalysis of the previous scheme and the architecture, it proposed an ECC-based privacy-preserving authentication, authorization, and key agreement scheme for WSNs in 5G-integrated IoT.

5.1 Consumption Overheads

This protocol, combined with the Kerberos authentication framework, has advantages over the elliptic curve cryptosystem based on single sign-on in the Li et al. scheme [21]. The parameters used in these two protocols are shown in Table 3.

Table 3: Comparison of the parameter length in our protocol with the Li et al. scheme [21]

Parameters used in this protocol	Length/B	Parameters in the Li et al. scheme [21]	Length/B
Device identity	16	Device identity	6
PUF challenge value	24	PUF challenge value	24

(Continued)

Table 3 (continued)

Parameters used in this protocol	Length/B	Parameters in the Li et al. scheme [21]	Length/B
PUF response value	24	PUF response value	24
Session key	16	Server private key	24
Hash value (SHA-256)	32	Hash value	24
TGT	48	Elliptic curve point	48
SGT	64	Synchronizing sequence	6
nonce	16	Random number n	4

The computational overhead is based on the computational overhead of various common password operations provided in [27], as shown in Table 4.

Table 4: Computational overhead of common encrypting operations

Symbol	Cryptography operation instructions	Computational overhead/ms
T_h	Unidirectional hash function operation	0.025
T_e	Encryption operation	0.087
T_d	Decryption operation	0.087
T_m	Elliptic curve point operation	1.287
T_p	Bilinear pair operation	3.298
T_{puf}	PUF operation	0.12
T_f	Fuzzy extraction	0.525

Based on the above parameters, the total computational and communication consumption (for a single login) for each scheme can be obtained. As shown in Table 5, it is clear that the computational and communication overheads for a vehicle's single login and a single APP in our proposed protocol are marginally superior to those in the protocols proposed by Chatterjee et al. [20] and Li et al. [21]. The computational overheads for a vehicle's multiple logins are depicted in Fig. 11.

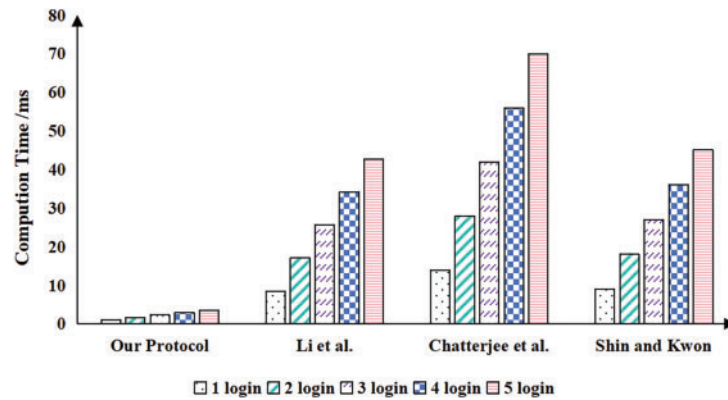
Table 5: Comparison of resource consumption schemes during the single login process

Protocols	Node	Computational overhead	Communication overhead/bit
Our protocol	V	$3T_h + 1T_e + 2T_d + 1T_{puf}$	2048
	AS	$3T_h + 2T_e$	960
	TGS	$2T_e + 2T_d$	768
	APP	$1T_d$	–
	Total	$6T_h + 5T_e + 5T_d + 1T_{puf} = 1.14 \text{ ms}$	3776
Li et al. [21]	Client1	$4T_h + 2T_m + 2T_{puf}$	1248
	Server	$6T_h + 2T_m$	4368
	Client2	$4T_h + 2T_m + 2T_{puf}$	1152
	Total	$14T_h + 6T_m + 4T_{puf} = 8.552 \text{ ms}$	6768
Chatterjee et al. [20]	Client1	$4T_h + 2T_p + 2T_{puf}$	–
	Server	$6T_h$	–

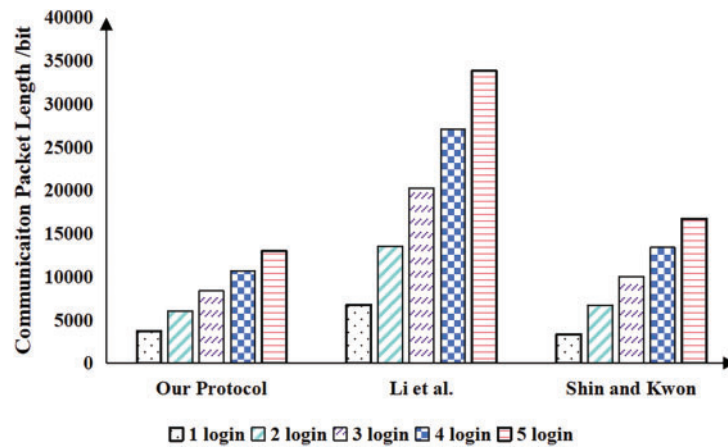
(Continued)

Table 5 (continued)

Protocols	Node	Computational overhead	Communication overhead/bit
Shin and Kwon [26]	Client2	$4T_h + 2T_p + 2T_{puf}$	–
	Total	$14T_h + 4T_p + 4T_{puf} = 14.002 \text{ ms}$	–
	User	$T_f + 14T_h + 3T_m$	1158
	AAS	$12T_h + T_m$	1516
	GW	$2T_m + 5T_h$	678
	Total	$31T_h + 6T_m + T_f = 9.022 \text{ ms}$	3352

**Figure 11:** Comparison of computational overheads for a vehicle during multiple login processes

While the communication overhead in our protocol is higher than that of the protocol put forward by Shin and Kwon [26], in the complex VANET environment where vehicles frequently log in to the APP, our protocol demonstrates enhanced efficiency. The communication overheads for a vehicle's multiple logins are depicted in Fig. 12.

**Figure 12:** Comparison of communication overheads for a vehicle during multiple login processes

Through the comparison of resource consumption of multiple APP in Figs. 11 and 12, it is evident that our protocols offer advantages in complex network environments. In our protocols, once the vehicle successfully authenticates with the AS and obtains a TGT, it only needs to send a SGT request to the TGS multiple times within the TGT's lifetime. Meanwhile, the APP having already authenticated the AS and obtained the shared key with the TGS, does not need to send messages to the vehicle or the KDC during the authentication phase. This significantly reduces the computational and communication overhead during vehicle authentication. The results of the analysis demonstrate that, compared to the scheme proposed by Li et al., our protocol achieves an 86.67% improvement in computational efficiency; compared to Chatterjee et al.'s scheme, it improves by 91.86%; and compared to Shin and Kwon's scheme, it improves by 87.36%.

5.2 Simulation Experiment

To verify the analysis, simulation environments of OMNeT 5.6.1++, SUMO 1.5.0 and Veins 5.0 are used. Where OMNeT++ is used for network simulation, SUMO is an open-source traffic emulator, and Veins is a framework for simulating the Internet of vehicles, including protocols and infrastructure. The specific model architecture of the Internet of Vehicles simulation platform is shown in Fig. 13.

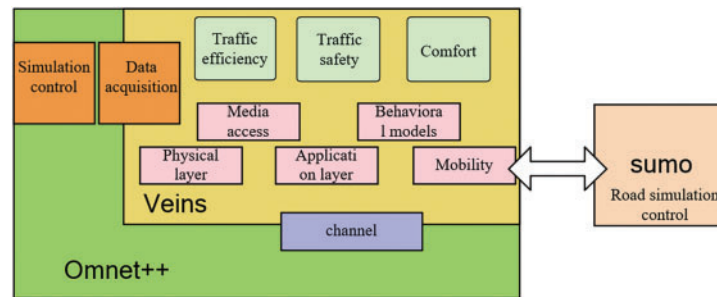


Figure 13: Simulation architecture diagram of the Internet of Vehicles

As shown in Fig. 14, the traffic network and vehicle movements are modeled in SUMO, including road networks, traffic signals, and vehicle traffic. A simulation area of 7×7 km around the campus was selected, and the vehicle and road data were randomly generated. Vehicle communication was then simulated in OMNeT++ using the Veins framework, which integrates vehicle motion data from SUMO with the communication model of OMNeT++ to simulate communication between vehicles and between vehicles and servers. Finally, the simulation results were analyzed in OMNeT++ to evaluate the performance of the communication protocol. Parameters used in the simulation are listed in Table 6.

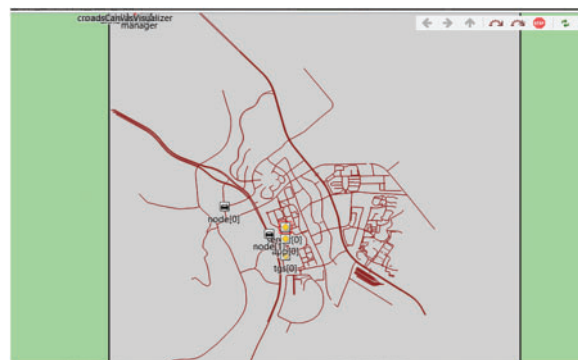
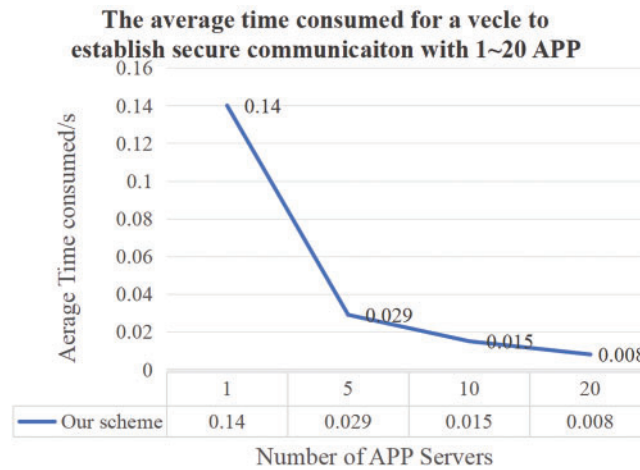


Figure 14: Simulation network diagram of the VANET

Table 6: Parameters used in simulation

Simulation parameters	Parameter values
Simulation area size	7×7 km
Simulation time	200 s
Number of vehicles	1–5
Number of APP	1–20
Vehicle speed	20 m/s
Communication bandwidth	6 Mbit/s
MAC protocol	IEEE 802.11p

The processes of authenticating and communicating between one vehicle and 1~20 APPs are simulated. The average time consumed for authenticate and key negotiation between a vehicle and an application server is shown in Fig. 15.

**Figure 15:** The average time consumed for authenticate and key agreement between a vehicle and an application server

In the PuKE-V2N protocol, it can be observed that the vehicle is required to perform mutual authentication solely with the AS initially. Subsequently, for each interaction with an application server, the authentication and key negotiation processes are streamlined. Instead of complex direct negotiations with the application server, the vehicle engages with the TGS. Through this interaction with the TGS, the vehicle can obtain the session key necessary for communication with the application server. As a result, as the number of application servers increases, the average time spent on identity authentication and key negotiation per interaction decreases. In terms of latency, the protocol significantly reduces communication rounds between vehicles and multiple application servers by leveraging a centralized TGS to generate and distribute session keys. As the number of application servers increases, vehicles only need to perform a single identity authentication with the AS to obtain a TGT, thereby avoiding redundant and complex authentication processes and lowering the average interaction delay. For throughput, the hierarchical authentication architecture (AS-TGS-APP) decouples authentication from service communication. The TGS enhances overall system throughput by enabling parallel processing of multiple vehicle requests, which effectively supports high-concurrency scenarios. Regarding response time, the protocol employs pre-configured CRPs and a rapid

Hash verification mechanism to minimize real-time computational overhead during the authentication phase, thereby reducing service acquisition latency. This protocol design ensures that vehicles swiftly establish secure sessions with application servers while maintaining robust security guarantees. This implies that the PuKE-V2N protocol becomes more efficient in handling multiple application servers, enabling faster and more streamlined communication between the vehicle and various application servers in the network.

6 Conclusion

This paper proposes a lightweight mutual authentication and key agreement protocol, PuKE-V2N, for the V2N VANET. First, the protocol replaces the password-based authentication mechanism in the standard Kerberos protocol with the CRPs generated by the vehicle's PUF. This substitution significantly fortifies the security posture by eliminating vulnerabilities associated with traditional password-based authentication. Second, the protocol capitalizes on the lightweight mutual authentication capabilities inherent in the Kerberos framework. This enables the efficient establishment of session key negotiation between the vehicle and the application server. By leveraging these features, the PuKE-V2N protocol optimizes the authentication process, ensuring seamless and secure communication. Finally, to evaluate the performance of the PuKE-V2N protocol, an Internet of Vehicles simulation environment was meticulously constructed. The evaluation results clearly indicate that when a vehicle engages with multiple application servers, the proposed scheme effectively reduces the average authentication time, thereby enhancing overall authentication efficiency. Future research endeavors will center on two critical aspects: integrating blockchain technology into the authentication process to enhance user privacy protection, and developing effective strategies for tracing malicious users. These efforts aim to further enhance the security and integrity of the V2N ecosystem.

Acknowledgement: We thank the School of Network Security of Jinling Institute of Technology for the support to this work.

Funding Statement: This work was supported in part by the Jiangsu “Qing Lan Project”, Natural Science Foundation of the Jiangsu Higher Education Institutions of China (Major Research Project: 23KJA520007), and Postgraduate Research & Practice Innovation Program of Jiangsu Province (No. SJCX25_1303).

Author Contributions: The individual contributions of the authors are as follows: Yanan Liu and Lei Cao: Led the research design, protocol development, and manuscript writing. Zheng Zhang: Conducted theoretical analysis and performance evaluations. Ge Li: Provided critical revisions. Shuo Qiu: Provided technical validation. Suhao Wang: Supervised the project and secured funding. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Manasrah A, Yaseen Q, Al-Aqrabi H, Liu L. Identity-based authentication in VANETs: a review. *IEEE Trans Intell Transp Syst.* 2025;26(4):4260–82. doi:10.1109/TITS.2025.3528932.
2. Ansari K. Joint use of DSRC and C-V2X for V2X communications in the 5.9 GHz ITS band. *IET Intell Transp Syst.* 2021;15(2):213–24. doi:10.1049/itr2.12015.
3. Fouda A, Berry R, Vukovic I. HARQ retransmissions in C-V2X: a BSM latency analysis. In: *Proceedings of the ICC 2024—IEEE International Conference on Communications; 2024 Jun 9–13; Denver, CO, USA.* doi:10.1109/ICC51166.2024.10622402.

4. Bi Y, Jia C. Towards resilience 5G-V2N: efficient and privacy-preserving authentication protocol for multi-service access and handover. *IEEE Trans Mob Comput.* 2025;24(6):5446–63. doi:10.1109/TMC.2025.3532120.
5. Bozorgzadeh E, Barati H, Barati A. 3DEOR: an opportunity routing protocol using evidence theory appropriate for 3d urban environments in VANETs. *IET Commun.* 2021;14(22):4022–28. doi:10.1049/iet-com.2020.0473.
6. Khah SA, Barati A, Barati H. A dynamic and multi-level key management method in wireless sensor networks (WSNs). *Comput Netw.* 2023;236(4):1–21. doi:10.1016/j.comnet.2023.109997.
7. Chen WW, Cao L, Shao CH. Blockchain based efficient anonymous authentication scheme for IOV. *J Comput Appl.* 2020;40(10):2992–9. (In Chinese). doi:10.11772/j.issn.1001-9081.2020020211.
8. Altaf F, Maity S. PLHAS: privacy-preserving localized hybrid authentication scheme for large scale vehicular ad hoc networks. *Veh Commun.* 2021;30(2):100347. doi:10.1016/j.vehcom.2021.100347.
9. Mukathe KD, Wu D, Ahmed W. Secure and efficient blockchain-based certificateless authentication scheme for vehicular *Ad-Hoc* networks (VANETs). In: *Proceedings of the 2022 4th International Conference on Applied Machine Learning (ICAML)*; 2022 Jul 23–25; Changsha, China. doi:10.1109/ICAML57167.2022.00065.
10. Li X, Jing T, Li R, Li H, Wang X, Shen D. BDRA: blockchain and decentralized identifiers assisted secure registration and authentication for VANETs. *IEEE Internet Things J.* 2023;10(14):12140–55. doi:10.1109/JIOT.2022.3164147.
11. Guo R, Dong R, Li X, Zhang Y, Zheng D. DRCLAS: an efficient certificateless aggregate signature scheme with dynamic revocation in vehicular *ad-hoc* networks. *Veh Commun.* 2024;47(2):100763. doi:10.1016/j.vehcom.2024.100763.
12. Neuman BC, Ts'o T. Kerberos: an authentication service for computer networks. *IEEE Commun Mag.* 1994;32(9):33–8. doi:10.1109/35.312841.
13. Qatinah SH, Al-Baltah IA. Kerberos protocol: security attacks and solution. In: *Proceedings of the 2024 1st International Conference on Emerging Technologies for Dependable Internet of Things (ICETI)*; 2024 Nov 25–26; Sana'a, Yemen. doi:10.1109/ICETI63946.2024.10777133.
14. Tung B, Neuman C, Hur M, Medivinsky A, Medivinsky S, Wray J, et al. Public key cryptography for initial authentication in Kerberos. In: *Proceedings of the Lecture Notes in Computer Science*; 2001; Berlin, Germany. doi:10.1007/11967668_24.
15. Sutradhar MR, Sultana N, Dey H, Arif H. A new version of Kerberos authentication protocol using ECC and threshold cryptography for cloud security. In: *Proceedings of the 2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*; 2018 Jun 25–29; Piscataway, NJ, USA. doi:10.1109/ICIEV.2018.8641010.
16. Liang YW, Zhu TP, Liu CH, Chen YF, Yang Y. Analysis of hadoop security mechanism based on kerberos security model optimization. In: *Proceedings of the 2024 International Conference on Telecommunications and Power Electronics (TELEPE)*; 2024 May 29–31; Frankfurt, Germany. doi:10.1109/TELEPE64216.2024.00060.
17. Maes R. *Physically unclonable functions: constructions, properties and applications*. Berlin/Heidelberg, Germany: Springer; 2013. doi:10.1007/978-3-642-41395-7.
18. Rai VK, Tripathy S, Mathew J. LPA: a lightweight PUF-based authentication protocol for IoT system. In: *Proceedings of the 2023 IEEE 22nd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*; 2023 Nov 1–3; Exeter, UK. doi:10.1109/TrustCom60117.2023.00233.
19. Hatti K, Paramasivam C. The MUX-based PUF architecture for hardware security. In: *2021 International Conference on Circuits, Controls and Communications (CCUBE)*; 2021 Dec 23–24; Bangalore, India. doi:10.1109/CCUBE53681.2021.9702737.
20. Chatterjee U, Chakraborty RS, Mukhopadhyay D. A PUF based secure communication protocol for IoT. *ACM Trans Embed Comput Syst.* 2017;16(3):1–25. doi:10.1145/3005715.
21. Li SS, Huang YC, Yu B, Bao BW. A PUF-based low cost secure communication scheme for IoT. *Acta Electron Sin.* 2019;47(4):812–7. doi:10.3969/j.issn.0372-2112.2019.04.007.
22. Idriss TA, Idriss HA, Bayoumi MA. A lightweight PUF-based authentication protocol using secret pattern recognition for constrained IoT devices. *IEEE Access.* 2021;9:8054–58. doi:10.1109/ACCESS.2021.3084903.
23. Liang W, Xie S, Zhang D, Bao BW. A mutual security authentication method for RFID-PUF circuit based on deep learning. *ACM Trans Internet Technol.* 2021;22(2):34. doi:10.1145/3426968.

24. Lai C, Wang X, Zheng D. A PUF-based authentication and key distribution scheme for in-vehicle network. In: Proceedings of the ICC 2023—IEEE International Conference on Communications; 2023 May 28–Jun 1; Rome, Italy. doi:10.1109/ICC45041.2023.10279633.
25. Tan WJ, Yang YT, Niu K, Qiao YX. MI-PUF-based secure authentication protocol for V2X communication information network security. Appl Res Comput/Jisuanji Yingyong Yanjiu. 2023;23(12):38–48. (In Chinese). doi:10.3969/j.issn.1671-1122.2023.12.005.
26. Shin S, Kwon T. A privacy-preserving authentication, authorization, and key agreement scheme for wireless sensor networks in 5G-integrated Internet of Things. IEEE Access. 2020;8(2):58675–89. doi:10.1109/ACCESS.2020.2985719.
27. Zhu QS. Research on security authentication in UAV ad hoc networks [master's thesis]. Nanjing, China: Nanjing University of Posts and Telecommunications; 2023. (In Chinese). doi:10.27251/d.cnki.gnjdc.2023.000193.