electronics

MDPI

Article

Enhancing Security of Automotive OTA Firmware Updates via Decentralized Identifiers and Distributed Ledger Technology

Ana Kovacevic [1,*] and Nenad Gligoric [2]

[1] Zentrix Lab, Research and Development Department, Milosa Trebinjca 12, 26000 Pancevo, Serbia
[2] Zentrix Lab, Blockchain Development Department, Milosa Trebinjca 12, 26000 Pancevo, Serbia; nenad.gligoric@zentrixlab.eu
* Correspondence: ana.kovacevic@zentrixlab.com

Abstract: The increasing connectivity and complexity of automotive systems require enhanced mechanisms for firmware updates to ensure security and integrity. Traditional methods are insufficient for modern vehicles that require seamless over-the-air (OTA) updates. Current OTA mechanisms often lack robust security measures, leaving vehicles vulnerable to attacks. This paper proposes an innovative approach based on the use of decentralized identifiers (DIDs) and distributed ledger technology (DLT) for secure OTA firmware updates of on-vehicle software. By utilizing DIDs for unique vehicle identification, as well as verifiable credentials (VCs) and verifiable presentations (VPs) for secure information exchange and verification, the solution ensures the integrity and authenticity of software updates. It also allows for the revocation of specific updates, if necessary, thereby improving overall security. The security analysis applied the STRIDE methodology, which enabled the identification of potential threats, including spoofing, tampering, and privilege escalation. The results showed that our solution effectively mitigates these threats, while a performance evaluation indicated low latency during operations.

Keywords: decentralized identifiers; distributed ledger technology; over-the-air updates; automotive security; verifiable credentials; vehicle authentication

Citation: Kovacevic, A.; Gligoric, N. Enhancing Security of Automotive OTA Firmware Updates via Decentralized Identifiers and Distributed Ledger Technology. *Electronics* **2024**, *13*, 4640. https://doi.org/10.3390/electronics13234640

Academic Editors: Antonio José Calderón Godoy, Isaías González Pérez and Francisco Javier Folgado Gaspar

Received: 14 October 2024
Revised: 11 November 2024
Accepted: 20 November 2024
Published: 25 November 2024

1. Introduction

The increasing connectivity and complexity of automotive systems bring many benefits but also significant security challenges. Modern vehicles are now frequently equipped with features that allow them to connect to Wi-Fi or mobile networks, enabling inter-vehicle communication and access to digital services such as real-time navigation, automatic software updates, and remote diagnostics [1]. Automotive systems are becoming more sophisticated and interconnected, providing advanced functionalities such as OTA updates, remote control, and diagnostics. OTA updates are particularly valuable as they allow manufacturers to efficiently fix software bugs, improve vehicle performance, and introduce new features without the need for physical servicing. OTA updates allow manufacturers to address security vulnerabilities in near real-time, significantly reducing the opportunity for attackers to exploit known issues. As vehicles become increasingly autonomous, even minor software bugs could result in catastrophic consequences, including loss of life or massive financial liabilities for OEMs. Traditional vehicle recalls due to software malfunctions are often costly and time-consuming, whereas many of these issues could be avoided or minimized using OTA updates. For example, General Motors recalled 4.3 million vehicles due to a software issue that prevented airbags from deploying [2], Honda recalled 350,000 vehicles due to a software glitch in the parking brake [3], and Jaguar Land Rover recalled more than 65,000 Range Rover sport utility vehicles due to a software bug that could cause vehicle doors to unlatch unexpectedly [4].

However, OTA advancements come with a significant trade-off: as vehicles become more connected, their attack surface expands, making them increasingly vulnerable to cyber

threats. Attacks on systems such as Kia, Hyundai, and Nissan, which allowed attackers to remotely control vehicles, demonstrated how vulnerable current remote communication methods are. An even more severe incident occurred with the Spireon system, where attackers gained administrative access to control 15.5 million vehicles, including remotely starting, shutting down, and updating firmware [5]. Similarly, Fiat Chrysler recalled millions of Jeep Cherokee vehicles due to a vulnerability in the infotainment system, which allowed attackers to remotely control the vehicles [6]. This vulnerability was exploited by researchers who demonstrated how attackers could remotely infiltrate the vehicle's head unit, bypass security mechanisms, and reprogram the gateway chip to inject arbitrary CAN messages. By doing so, they gained control over critical physical functionalities, such as steering and braking, even while the vehicle was in motion. This attack highlights the grave implications of weak code verification processes in automotive systems and the lack of anomaly detection on the CAN bus, which could have mitigated or prevented such exploits.

Furthermore, one of the most commonly used authentication mechanisms in OTA systems is public key infrastructure (PKI). Although PKI provides a strong foundation for authentication and data integrity, it also poses a risk due to its centralized nature, where the compromise of a central certificate authority (CA) can lead to a single point of failure, bottlenecks and challenges in managing trust relationships [7]. Related incidents that highlight these weaknesses include the Lenovo Superfish incident [8] and Diginotar's "Operation Black Tulip" [9]. The Lenovo Superfish incident in 2014 highlighted risks in centralized PKI systems, where pre-installed adware included a non-unique, trusted root CA certificate, enabling attackers to spoof HTTPS traffic and create fraudulent certificates for sensitive websites. Similarly, the DigiNotar "Operation Black Tulip" breach in 2011 involved the compromise of a CA server, allowing the issuance of rogue certificates, including one for Google, which was used in Iran for intercepting secure communications. Both incidents underscore the dangers of centralized PKI, where a single compromised CA can pose global security risks due to weak password policies, outdated software, and inadequate network segmentation.

To overcome these issues, a solution based on decentralized technologies such as DIDs and DLT has been proposed. By using DIDs for unique identification and VCs for secure and verifiable communication, the proposed solution enhances the security of OTA updates, eliminating centralized points of vulnerability. Additionally, hash functions are employed to ensure data integrity, allowing vehicles to verify that firmware updates have not been tampered with during transmission. This approach ensures robust authentication, reduces reliance on centralized systems prone to single points of failure, and enhances the transparency of the update process. By incorporating a revocation mechanism, the solution allows manufacturers to promptly address security breaches or outdated credentials. Together, these features provide a scalable and resilient framework for secure OTA updates in connected vehicles.

This paper is organized as follows: Section 2 provides the background and related work; Section 3 introduces the proposed solution for secure OTA updates; Section 4 illustrates the implementation of the proposed solution, focusing on the key steps; Section 5 analyzes potential threats using the STRIDE methodology and evaluates the solution's key performance indicators; Section 6 discusses the implications and advantages of the proposed approach; and Section 7 presents conclusions and recommendations for future research.

2. Background and Related Work

Firmware updates are typically performed when the vehicle is not in use, such as overnight or when parked, to avoid potential risks and disruptions while driving. The duration depends on the update size, with larger or multi-module updates requiring more time, especially for complex functionalities or security patches [10]. Initially, OTA technology was limited to basic software updates (SOTA), but it has since evolved to include firmware over-the-air (FOTA) updates, enabling manufacturers to enhance critical

hardware components without physical intervention. OTA updates can be distributed by either software suppliers (Tier 1) or directly by the original equipment manufacturer (OEM). In cloud-based OTA architecture, vehicle manufacturers upload software updates to a cloud server, where connected vehicles can access and download the latest software versions [11]. OTA technology has been extensively researched in the context of automotive systems, with various approaches proposed to enhance its efficiency and security.

Kent et al. [12] propose a security framework using PKI and public key cryptography (PKC) for the authenticity and integrity of firmware updates. Their solution involves leveraging the existing relationship between OEMs and Tier-1 suppliers, using a split-signing authority model. In this model, both the supplier and the manufacturer sign the firmware to ensure that updates are not tampered with during delivery.

Manna et al. [13] propose a solution using attribute-based encryption (ABE) to secure OTA firmware updates. Their approach focuses on maintaining data confidentiality during transmission and storage. They address the challenge of encrypting updates so that only authorized electronic control units (ECUs) can decrypt and apply the updates, even if an adversary gains access to the vehicle's gateway or cloud server.

Ghosal et al.'s [14] STRIDE guarantees end-to-end security by utilizing ciphertext-policy attribute-based encryption (CP-ABE), which provides fine-grained access control. Only vehicles that meet specific conditions, as defined by the encryption policy, can access the software updates. This approach ensures confidentiality, addressing a critical gap in many existing solutions that typically focus only on authenticity and integrity. The updates are stored in the cloud, where in-network storage and replication are utilized to minimize data access time.

Plappert et al. [15] propose a mechanism for distributing OTA software updates in connected vehicles using Trusted Platform Module (TPM) 2.0. The TPM serves as a hardware root of trust for managing cryptographic processes, with asymmetric cryptography used for the external backend, while it translates to symmetric cryptography within the vehicle to reduce network load.

Seo et al. [16] proposed the use of blockchain technology as a secure method for storing firmware data and maintaining immutable records of all updates. UAVs are employed to distribute firmware updates to devices without internet connectivity, transporting updated firmware files to IoT devices in remote locations. The system utilizes a private blockchain network Hyperledger platform, where participants (IoT devices, manufacturers, and UAVs) are registered through a verification process. Public-key encryption and Bloom filters are employed to verify UAVs and IoT devices, ensuring that updates originate from legitimate sources.

Baza et al. [17] proposed a blockchain-based distributed firmware update scheme specifically tailored for autonomous vehicles (AVs). The proposed solution uses a consortium blockchain of different AV manufacturers to ensure the authenticity and integrity of firmware updates. It employs attribute-based encryption (ABE) to control access and zero-knowledge proofs (zk-SNARK) for secure proof of distribution among AVs without revealing sensitive information. A reward system is established to incentivize AVs to distribute updates, ensuring high availability and fast delivery.

Oham et al. [18] introduce the B-FERL framework for securing smart vehicles. The framework uses a permissioned blockchain to manage access to vehicle data, ensuring that only trusted entities can participate. The system monitors the internal state of vehicles through a challenge–response mechanism, particularly focusing on electronic control units (ECUs). This mechanism checks for compromised or altered vehicle networks.

Choi and Lee [19] propose a distributed architecture for firmware updates for IoT devices based on the SUIT (Software Updates for Internet of Things) architecture, utilizing blockchain. The proposed architecture employs nodes for firmware registration and retrieval, where firmware is stored in a distributed file system, while hash values and manifests are recorded on the blockchain. This approach addresses issues such as the

author-disappearing problem and resilience to single points of failure, enabling irreversible and secure updates.

Despite the significant contributions of the aforementioned research to improving OTA update security, most existing solutions remain heavily dependent on centralized or static security mechanisms, such as attribute-based encryption (ABE) or traditional blockchain protocols. The reliance on centralized architectures introduces single points of failure and creates bottlenecks that can compromise the system's efficiency and reliability under high-demand scenarios. Moreover, there is a noticeable gap in comprehensive solutions that specifically address the authentication challenges inherent to OTA updates. Authentication is critical to ensuring that firmware updates are genuine, tamper-proof, and delivered only to authorized devices. Current methodologies often focus on securing the transmission of updates but fail to provide robust mechanisms for verifying firmware origins or ensuring traceability throughout the update lifecycle.

Our proposed approach bridges these gaps by integrating DIDs and VCs with DLT. The proposed solution enables dynamic and decentralized verification of firmware identity and integrity, significantly reducing the risks associated with traditional, centralized vulnerabilities. By leveraging DLT's immutability and transparency, the model ensures secure, verifiable interactions between manufacturers, vehicles, and other stakeholders. Additionally, the use of DIDs and VCs enhances trust by providing a flexible mechanism for managing authentication, firmware versioning, and revocation processes.

3. Proposed Solution

In the following sections, the proposed solution is outlined, focusing on core components. The System Architecture Overview section presents the overall architecture, detailing the layers involved in secure OTA firmware updates using DIDs and DLT technologies. Next, the Key Components section provides a breakdown of each component, including identity management and DLT for data integrity and firmware verification.

3.1. System Architecture Overview

To address the growing security challenges in traditional centralized OTA solutions, this research proposes the use of DIDs, VCs, and DLT for secure software updates in vehicles. DIDs provide unique identification for each vehicle and OEM, while VCs ensure that vehicle data and updates come from a trusted source and allow for additional information about involved entities to be shared.

The proposed solution consists of two layers: an identity management layer and a DLT layer for firmware validation, as shown in Figure 1. Each vehicle's DID is registered on the IOTA Tangle network and is used for authentication and cryptographic key exchange between the vehicle and the OEM.

The OTA update process includes downloading the firmware and then installing it when the vehicle is parked and not performing any active functions like charging or driving. When a new firmware version is available, its hash is published on the DLT network. A hash, generated by a cryptographic hash function like SHA-256 or MD5, converts input data into a fixed-length output known as a hash value [20]. The unique property of a hash function is that it yields the same output for the same input but changes entirely with even the smallest alteration, making it ideal for verifying data integrity.

Before downloading, the vehicle verifies the source using the OEM's VC to ensure that the update originates from a valid source. After successful authentication, the vehicle can download the firmware. Once the download is complete, the vehicle calculates the file hash to confirm that it has not been altered. If the hash values match, the vehicle proceeds with installation. Since there may be a time gap between the download and installation, hash verification ensures that the file has not been modified and can be safely installed. Even if an attacker intercepts the encrypted communication during transmission, the vehicle will reject the update if the hash values do not match, ensuring the authenticity and integrity of the update.

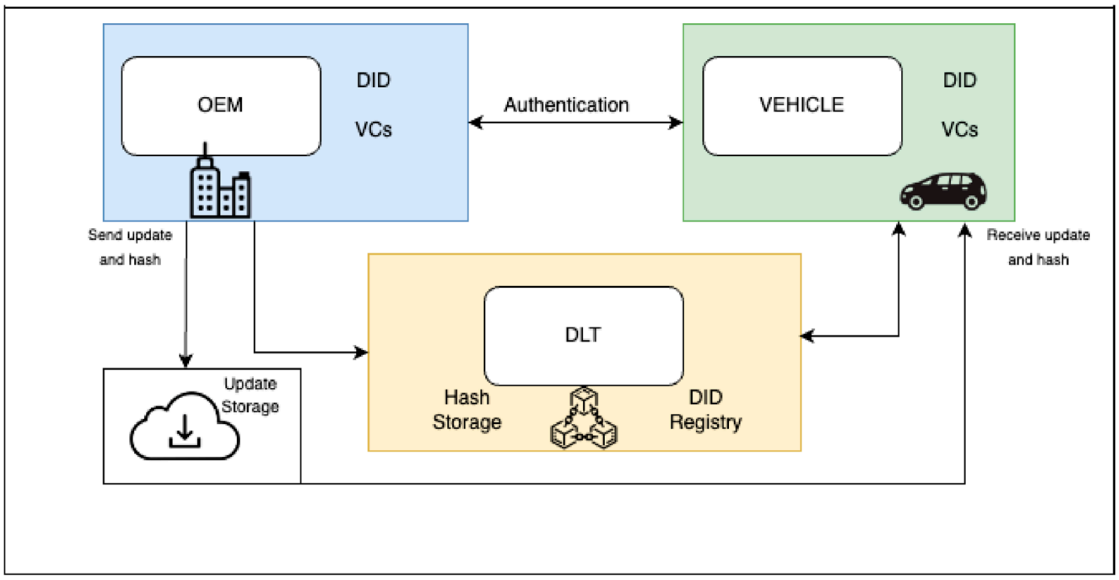


Figure 1. High-level architecture of the proposed solution.

3.2. Key Components

The following subsections provide a detailed overview of each key component and its role in the overall architecture of the solution.

3.2.1. Distributed Ledger Technology

DLT represents a decentralized system for recording data, where each node maintains its copy of the ledger, ensuring transparency and decentralization in data management. This approach significantly reduces the risk of data compromise, as any change requires the consensus of the majority of participants. DLT operates in a multi-party environment without a central authority, enabling functionality even in the presence of untrusted participants [21]. This enhances resilience against manipulations and eliminates single points of failure, making it suitable for applications that require high levels of trust and security. DLT networks can be classified as public or permissionless (such as Bitcoin, Ethereum, and Cardano) and private or permissioned (such as Quorum and Hyperledger). Each type of network has its advantages and disadvantages in terms of scalability, security, and decentralization. For instance, public networks like Ethereum often incur high transaction fees and have lower scalability, while private networks offer greater control over access but are perceived as less secure compared to permissionless networks [22].

Traditional blockchain systems can pose challenges for implementations such as the Internet of Vehicles (IoV) due to high operational costs, energy consumption, and limited transaction throughput, making them less suitable for applications that require high scalability. An alternative solution is DLTs that utilize a directed acyclic graph (DAG) structure, like IOTA. Unlike conventional blockchains, IOTA records transactions directly into the ledger, allowing for parallel processing and higher throughput, which effectively resolves the scalability issues inherent to the blockchain. IOTA eliminates the need for mining, enabling feeless transactions and reducing energy consumption [23]. Although early versions of IOTA faced criticism due to centralization around a special node known as the coordinator, IOTA 2.0 eliminates the coordinator, transitioning towards a fully decentralized system that enhances both scalability and security [24]. IOTA uses the UTXO (Unspent Transaction Output) model for managing transactions. This approach allows each output from a transaction to remain unspent until it is consumed in a new transaction, which enhances user security and privacy. Additionally, this model enables efficient resource management, as each transaction generates new outputs that can be utilized for future transactions. In the proposed solution, the IOTA protocol is utilized for storing hash values and managing identities, serving as a verifiable data registry for DIDs.

3.2.2. Decentralized Identifiers and Verifiable Credentials

DIDs, proposed by the W3C Consortium [25], enable entities to have full control over their digital identities without depending on third parties. Each identifier is linked to a unique document known as the DID document, which includes the associated pair of public and private keys, as well as information on verification and authentication methods. DIDs and DID documents are registered on the IOTA ledger. In addition to DIDs, VCs provide a cryptographically verifiable and tamper-proof record of claims (e.g., qualifications, attributes, or identities), which can be independently verified without direct access to the issuer [26]. VCs are linked to DIDs, offering more detailed information about the entity.

In the proposed solution, VCs are used for two main purposes: (1) to verify the identity and rights of the OEM when sending updates to the vehicle, and (2) to include information about the vehicle's attributes, such as the firmware version and update eligibility. OEMs can issue VCs that specify the latest firmware details and installation permissions. After the update is downloaded and installed, the OEM provides a new VC containing information about the new firmware version, installation date, and compliance with the latest software standards. The system also supports a revocation mechanism, which ensures that outdated or compromised credentials are automatically invalidated, preventing rollback attacks or unauthorized access to the vehicle's system.

4. Implementation Overview

This section illustrates the implementation of the proposed solution, covering essential steps such as generating DIDs, creating and issuing VCs, and verifying VPs and hash transfer and validation.

Figure 2 illustrates the process of generating DIDs for the OEM and the vehicle, as well as creating a VC for the vehicle. During the initial registration, a unique DID is created to identify the vehicle on the network. Next, the OEM issues a VC that includes information such as the vehicle model, firmware version, and access rights. In the realm of vehicle management, VCs are essential for ensuring that each vehicle is recognized as compliant and authorized. Additionally, the VC incorporates a timestamp along with a mechanism for revocation.

```
Generated OEM DID:
did:iota:rms:0xd1908b907e49412aa74a6939acb5e2de06b37ec441cc635704687cebfa27a284


Generated Vehicle DID:
did:iota:rms:0xebaabe06baa7ff228faf8663f7ff7794c4c485fbd516f5200419d62877d9d401


Created VC: {
  '@context': 'https://www.w3.org/2018/credentials/v1',
  id: 'https://example.com/credentials/vehicle-info',
  type: [ 'VerifiableCredential', 'VehicleInfoCredential' ],
  credentialSubject: {
    id: 'did:iota:rms:0xebaabe06baa7ff228faf8663f7ff7794c4c485fbd516f5200419d62877d9d401',
    access_rights: 'download_update',
    firmware_version: '0.1',
    manufacture_year: '2022',
    status: 'active',
    vehicle_model: 'Model X'
  },
  issuer: 'did:iota:rms:0xd1908b907e49412aa74a6939acb5e2de06b37ec441cc635704687cebfa27a284',
  issuanceDate: '2024-10-07T04:30:24Z',
  credentialStatus: {
    id: 'did:iota:rms:0xd1908b907e49412aa74a6939acb5e2de06b37ec441cc635704687cebfa27a284#revocation-service',
    type: 'RevocationBitmap2022',
    revocationBitmapIndex: '5'
  }
}
```

Figure 2. DIDs generation and creation of VC for the vehicle.

There are several scenarios in which the OEM may need to revoke a vehicle's VC. One reason for revocation occurs when a new firmware version is installed. The existing VC

may become outdated, prompting the OEM to issue a new one that accurately reflects the vehicle's capabilities. Security vulnerabilities may also necessitate revocation. If a risk is identified that compromises the vehicle's systems, the OEM can revoke the VC to prevent unauthorized access. In cases of recalls due to defects or safety concerns, the VC can be revoked to ensure that the vehicle is not allowed to operate until the issues are resolved.

The OEM holds its own VC, which includes the OEM's DID and metadata related to the update. When the update is sent, the OEM provides its VC as part of a VP formatted as a JWT (JSON Web Token). This VP serves to confirm the OEM's identity and includes important information about the update. A VP allows one or more VCs to be securely shared in a way that ensures their authenticity and integrity, as shown in Figure 3.

```
OEM sends a Verifiable Presentation:
{
  '@context': 'https://www.w3.org/2018/credentials/v1',
  type: 'VerifiablePresentation',
  verifiableCredential: [

'eyJraWQiOiJkaWQ6aW90YTpybXM6MHhkMTkwOGI5MDdlNDk0MTJhYTc0YTY5MzlhY2I1ZTJkZTA2YjM3ZWM0N
DFjYzYzNTcwNDY4N2NlYmZhMjdhMjg0I2Q.....1hcEluZGV4IjoiNSJ9fX0.-grqh4Nh7u32tND1JcdnmELMFl9e
ASnUlV4hSfBYaRzm00SU0acFbvN2lHCiplF3ncnlwYaGJCs4otyb2Wo6Aw'
  ],
holder:
'did:iota:rms:0xd1908b907e49412aa74a6939acb5e2de06b37ec441cc635704687cebfa27a284'
}
```

Figure 3. VP sent by the OEM.

When the vehicle receives the update accompanied by a VP, it verifies the VP by checking its digital signature and ensuring that the included VC is valid. If verification is successful, the vehicle proceeds to download the update. Figure 4 depicts the sequence of steps involved in verifying the VP. This process ensures that the OEM's identity and the integrity of the update are validated using the DID and VC. If the VP is verified successfully, the vehicle proceeds to the next step in the update process, such as querying the hash value for the firmware. Otherwise, the process is terminated to ensure the integrity of the system.

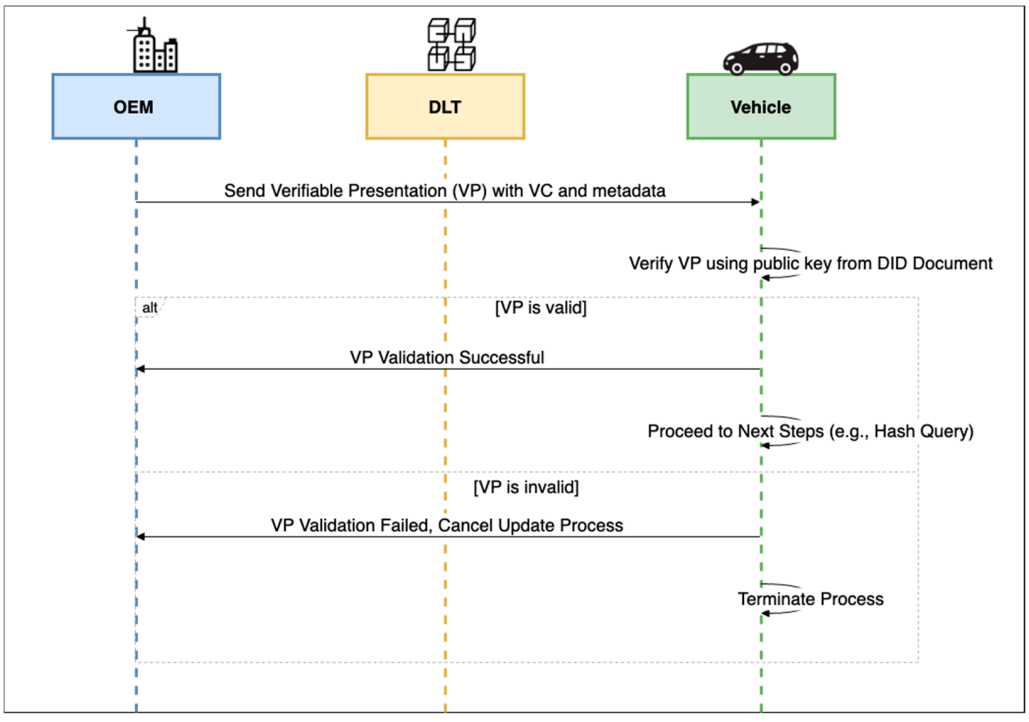


Figure 4. Sequence diagram for verifying VP.

After downloading, the vehicle calculates the hash of the downloaded firmware and compares it with the hash value stored on the DLT network. If the hash values match, the vehicle proceeds with the installation process, as shown in Figure 5.

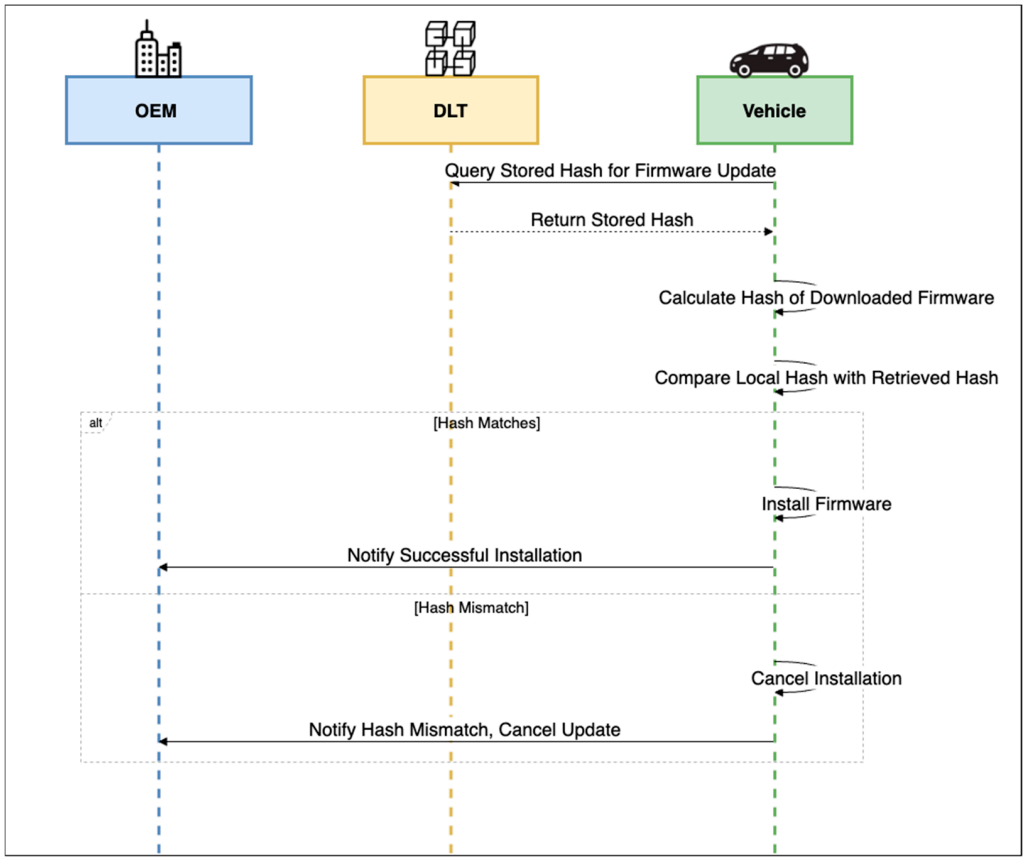
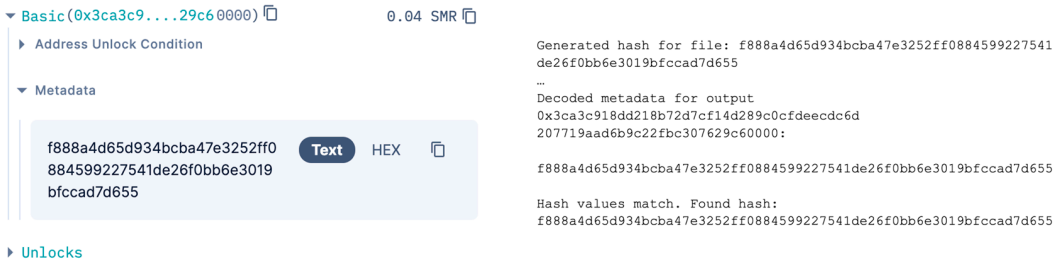


Figure 5. Sequence diagram for firmware hash validation.

The hash value is stored in the metadata field on the DLT network. Figure 6 illustrates the process of downloading and verifying update hash values. Figure 6a displays the section of the IOTA explorer where the metadata field containing the hash is shown. Figure 6b demonstrates how the vehicle compares its local firmware hash with the hash retrieved from the DLT network. Upon initiating the update download, the vehicle calculates the hash of the received firmware and checks it against the value stored on the DLT. If the values match, the vehicle will start with installation. After successful installation, the vehicle's VC is revoked, and a new one is generated with updated information.



(**a**) (**b**)

Figure 6. Process of downloading and verifying firmware hash values: (**a**) shows the metadata field in the IOTA explorer where the hash is displayed; (**b**) illustrates the vehicle comparing its local firmware hash with the hash retrieved from the DLT.

The VP, which includes the OEM's VC, ensures that the update is coming from a legitimate and trusted source. This protects against malicious updates. Hash verification confirms that the firmware has not been altered during transmission. Even if an attacker intercepts and modifies the firmware, the mismatch in hash values would immediately invalidate the update. The VP confirms the authenticity of the sender, while hash verification ensures data integrity, providing a two-layered security mechanism.

5. Security Analysis and Performance Evaluation

This chapter presents a security analysis using the STRIDE methodology and an evaluation of the performance of the proposed solution based on key metrics.

5.1. Security Analysis

The STRIDE methodology, developed by Microsoft, is a widely used framework for identifying security vulnerabilities in software systems [27]. It categorizes threats into six main areas (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege) and enables a structured assessment and implementation of appropriate security measures. Furthermore, STRIDE is considered a mature and well-established tool for threat modelling, extensively used in both industry and academic research, particularly within the automotive domain [28]. Table 1 summarizes the various types of attacks identified through the STRIDE methodology.

Table 1. Description of different attacks identified through the STRIDE methodology.

Threat	Description
Spoofing	An attacker can impersonate a legitimate user or system to gain unauthorized access.
Tampering	Unauthorized modification of data or software, often during transmission.
Repudiation	A user can deny performing an action, complicating accountability.
Information Disclosure	Leakage of private data during transmission or storage.
Denial of Service (DoS)	Attacks aimed at making a system or service unavailable to intended users.
Elevation of Privilege	An attacker gains unauthorized access to higher privileges within the system.

The following list outlines the potential attacks specific to our implementation, along with the strategies we have adopted to mitigate them:

- Spoofing—An attacker may impersonate a vehicle, server, or user to perform unauthorized actions or download firmware. To mitigate this risk, DIDs are utilized for identity verification, and VCs are used to verify identities before any data exchange occurs.
- Tampering—An attacker may alter firmware during transmission from the cloud to the vehicle. To address this, the integrity of updates is verified using hash values on DLT. Additionally, the firmware is accompanied by a VP sent from the OEM. If the hash values do not match or if the VP verification fails, the installation is rejected, preventing any unauthorized modifications.
- Repudiation—Users may deny that certain actions were performed or that they consented to them. Since DLT is used, all actions are transparently logged. VCs are employed to confirm actions and ensure traceability.
- Information Disclosure—There is a risk of leakage of private vehicle data or firmware content during transmission or storage. Data must be encrypted during both transmission and storage. Only hash values are stored on the DLT, not the actual data.
- Denial of Service (DoS)—DoS attacks may attempt to disable access to services or information. By using DLT for the registration of DIDs, centralized points of vulnera-

bility are reduced. The DLT architecture increases system resilience and ensures that necessary identities and data remain accessible, even during potential attacks.

- Elevation of Privilege—An attacker may attempt to gain unauthorized access to privileged updates or server settings. Our solution includes a DID registry for vehicles and VCs to define access rights more granularly, along with role-based access control (RBAC) for effective privilege management.

5.2. Evaluation of Key Performance Indicators (KPIs)

In this section, we present a performance evaluation of the proposed solution, with an emphasis on latency as a critical performance metric. The evaluation focuses on measuring the time required for key operations, including creating and resolving DIDs, issuing and verifying VCs and VPs, and validating hash values. These metrics were selected to provide meaningful insights into the system's efficiency and its ability to address challenges related to authentication, data integrity, and response speed, all of which are essential for secure OTA updates.

The methodology used for this evaluation is centered on KPIs that are specifically relevant to the security and operational efficiency of over-the-air systems. These KPIs reflect the system's ability to handle real-time applications while maintaining robust security measures. By focusing on latency during critical operations, the evaluation demonstrates how the proposed solution aligns with the requirements for secure communications and timely responses in automotive environments. Additionally, the inclusion of these KPIs aligns with principles outlined in standards such as ISO/SAE 21434 [29], which emphasize the importance of efficient and secure data handling in vehicular systems.

Testing was conducted on a local machine and a cloud server. The local machine was a MacBook Pro with an Apple M3 Pro chip, 18 GB RAM, and macOS 14.3, used for generating and signing JWTs. The IOTA Hornet node was hosted on a DigitalOcean cloud server running Ubuntu 24.04 LTS, with 2 vCPUs and 8 GB RAM. The cloud server handled operations such as DID registration, VC validation, and hash value sending and retrieval. All requests to the IOTA node were sent via the API endpoint. Each operation was executed in 20 iterations, and operations were performed serially. The results of this testing are summarized in Table 2.

Table 2. Performance Evaluation of Latency for Operations in IOTA.

Operation	Description	Average Time (ms)
DID Registration	Time needed to register a new DID on IOTA network	17.772 (ms)
DID Resolution	Time needed to resolve a registered DID and retrieve DID Document	94.356 (ms)
VC Issuance	Time to sign and issue VC	1.72 (ms)
VC Creation	Time to generate a VP from an existing VC	1.75 (ms)
VP Verification	Time needed to verify a Verifiable Presentation (VP)	99.74 (ms)
Hash Retrieval	Time needed to retrieve and validate stored hash	363.895417 (ms)

The results of our study highlight the system's ability to efficiently manage operations involving DIDs, VCs and hash validation, demonstrating low latency across a range of critical tasks. The cloud-based IOTA node showcased reliable performance during testing, with average processing times well within acceptable limits for real-time applications. While the primary focus of this evaluation was latency, the broader capabilities of the IOTA network significantly enhance its suitability for large-scale and high-frequency operations. Testing conducted by the IOTA Foundation has shown that the network can support throughput of up to 500 blocks per second (BPS) with minimal latency [30]. This capability underscores its potential for managing large-scale operations, such as OTA updates for fleets of vehicles. In such scenarios, the ability to handle simultaneous data requests without bottlenecks is crucial.

For instance, Tesla determines the timing and nature of updates based on factors such as vehicle configuration, software versions, and user preferences [31]. While this approach provides flexibility and customization, it also introduces variability in update timing and duration. Updates may require significant time to complete, particularly for complex firmware, and the reliance on centralized servers can lead to delays during high-demand periods. These limitations emphasize the importance of a decentralized approach that can handle variability without compromising performance, especially in light of incidents like the 2016 Tesla hack [32], where vulnerabilities in centralized OTA mechanisms allowed attackers to remotely compromise critical vehicle systems.

As highlighted in Section 3.2.1., our choice of IOTA is motivated by its DAG structure, which provides better scalability and flexibility compared to traditional blockchain systems. The decentralized architecture of IOTA ensures resilience against common attacks such as Sybil and single-point-of-failure attacks, making it a robust solution for secure OTA updates [33]. This architecture enables horizontal scaling by allowing multiple transactions to be validated in parallel. This scalability advantage becomes increasingly evident as the network grows, as shown in the work of Kahmann et al. [34], where the DAG structure of IOTA was shown to improve both transaction throughput and latency as the number of network participants increased.

While there is limited work directly comparing systems using DIDs, VCs, and hashes for OTA updates, the selected KPIs align with real-world requirements for authentication, data integrity, and efficiency in vehicular communication systems. This makes the proposed solution a unique contribution to the field.

6. Discussion

The proposed solution improves upon existing OTA firmware update methods by addressing challenges such as centralization, scalability, and transaction costs. Traditional approaches, particularly those relying on PKI, are vulnerable to central points of failure, as they depend on centralized authorities. Even encryption-based methods, while effective at securing data, often involve inefficient key management due to central control points. Similarly, solutions based on hardware security or conventional blockchain systems frequently encounter high transaction fees and limited scalability. In contrast, the proposed approach leverages DIDs, VCs, and DLT, offering a decentralized model that eliminates these vulnerabilities. By assigning each vehicle a unique identifier via DIDs, and using VCs to verify and authenticate firmware updates, we ensure a tamper-proof and verifiable update process. The addition of hash functions further enhances the integrity of firmware, allowing vehicles to verify the authenticity of updates before installation. This model reduces the risk of single points of failure and strengthens overall security in comparison to centralized systems. The integration of these components into the OTA process represents a significant advancement over traditional methods. The decentralized architecture also supports greater resilience against malicious attacks, particularly Sybil and man-in-the-middle attacks, due to the cryptographic trust inherent in DLT. Although direct empirical comparisons with existing methods are not feasible due to the novelty of our approach, the proposed system demonstrates significant potential in addressing current challenges in OTA security. Unlike existing solutions, this approach uniquely integrates DIDs and VCs for decentralized authentication, filling gaps in current research by providing tamper-proof identity management and revocation mechanisms. The scalability of the proposed system in large-scale deployments involving thousands of vehicles is inherently supported by IOTA's DAG architecture, which enables parallel transaction processing and minimizes the risk of network congestion during simultaneous updates. However, at the application layer, adaptive scheduling techniques could further optimize performance, where updates are distributed based on predefined priorities, such as critical security patches being prioritized over feature updates.

Furthermore, the use of DIDs and VCs can be extended to other use cases in vehicle-to-everything (V2X) communication, where it has already been proposed as a solution for

authentication, with existing initiatives by organizations like the Mobility Open Blockchain Initiative (MOBI) [35]. For example, incorporating decentralized identity management into V2X could prevent scenarios like remote vehicle hijacking through browser vulnerabilities, as it would ensure multi-layered authentication and data integrity. For entity verification and presenting VCs in scenarios requiring privacy (e.g., location sharing in V2I communication), the proposed solution can be extended with zero-knowledge proofs (ZKPs) [36].

Thus, future research should focus on ensuring interoperability between different systems. In terms of cross-chain interoperability [37], which involves communication between blockchain networks, different protocols can be used. For example, LayerZero is already compatible with IOTA EVM [38]. This would enable the transfer of information and data verification across different blockchain networks, which is especially important in scenarios where vehicles or devices communicate with different blockchain platforms. To realize this potential, researchers should develop standard APIs and middleware layers to streamline communication across heterogeneous blockchain networks. These tools should also address issues such as data consistency, latency, and security risks inherent in cross-chain operations. Testing these implementations in simulation environments mimicking real-world multi-blockchain setups will be essential for validating their robustness. When it comes to interoperability with off-chain systems, it is necessary to explore the compatibility of DIDs and VCs with protocols like transport layer security (TLS). Existing proposals [39] for using DIDs with TLS provide a basis for further research, which could also include the development of standards for easier integration with existing networks and applications. In addition to cross-chain interoperability, research on standardization frameworks is needed to ensure seamless adoption. Furthermore, future research could further enhance scalability by exploring techniques such as sharding. Sharding [40] involves dividing the network into smaller, more manageable fragments (shards), allowing transactions to be processed in parallel across these groups rather than across the entire network. This approach significantly boosts network capacity and throughput, alleviates the burden on individual nodes, and facilitates more efficient management of large-scale networks with numerous participants.

7. Conclusions

In this paper, we have proposed a decentralized solution for secure OTA firmware updates in connected vehicles, utilizing DIDs for vehicle authentication, VCs for verifying update authenticity, and hash functions for ensuring firmware integrity. The system was evaluated using the STRIDE security framework, demonstrating its resilience against common attacks. Additionally, we conducted a performance evaluation, focusing on latency, which confirmed the system's efficiency in managing key operations such as DID registration and VC verification.

The main contributions of this study include the development of a secure, decentralized architecture for OTA updates, the integration of DIDs and VCs for robust authentication, and the demonstration of scalable system performance suitable for real-time automotive environments. These contributions directly address key challenges in centralization, security, and data integrity faced by traditional systems. The proposed solution offers a scalable and adaptable framework for enhancing automotive cybersecurity in the context of OTA updates.

Future research will focus on optimizing the system's performance through techniques such as sharding and enhancing interoperability with off-chain systems to support broader applications in the automotive industry. Specifically, upcoming work will explore cross-chain communication protocols and dynamic network reconfiguration methods to address scalability and operability challenges in heterogeneous environments. Furthermore, the potential of decentralized oracles to enable real-time external data validation will be evaluated, alongside the implementation of ZKPs for privacy-centric use cases.

Author Contributions: Conceptualization, N.G. and A.K.; methodology, N.G.; software, A.K.; validation, A.K. and N.G.; formal analysis, A.K.; investigation, N.G.; resources; writing—original draft preparation, N.G.; writing—review and editing, A.K.; visualization, A.K.; supervision, N.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded by the European Commission under the framework of Horizon Europe CONFIDENTIAL6G project (Grant Agreement No. 101096435).

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Abdelkader, G.; Elgazzar, K.; Khamis, A. Connected Vehicles: Technology Review, State of the Art, Challenges and Opportunities. *Sensors* **2021**, *21*, 7712. [CrossRef] [PubMed]
2. CBC News. General Motors Recalls 4.3 Million Vehicles Worldwide to Fix Airbag Software Defect. CBC. 2016. Available online: https://www.cbc.ca/news/business/general-motors-recall-airbag-software-1.3755030 (accessed on 14 October 2024).
3. Honda Recalls 350,000 Civics for Electronic Parking Brake Glitch. Autoweek. Available online: https://www.autoweek.com/news/a1856001/honda-recalls-350000-civics-electronic-parking-brake-glitch/ (accessed on 3 October 2024).
4. BBC News. Software Bug Prompts Range Rover Recall. Available online: https://www.bbc.com/news/technology-33506486#:~:text=Land%20Rover%20is%20recalling%20more,sold%20between%202013%20and%20now (accessed on 4 October 2024).
5. Curry, S. Web Hackers vs. the Auto Industry: Critical Vulnerabilities Found in Automotive Systems, Affecting over 15 Million Vehicles. Available online: https://samcurry.net/web-hackers-vs-the-auto-industry (accessed on 1 October 2024).
6. Miller, C. Lessons Learned from Hacking a Car. *IEEE Des. Test* **2019**, *36*, 7–9. [CrossRef]
7. Rathore, H.; Samant, A.; Jadliwala, M.; Mohamed, A. TangleCV: Decentralized Technique for Secure Message Sharing in Connected Vehicles. In Proceedings of the ACM Workshop on Automotive Cybersecurity, Richardson, TX, USA, 27 March 2019; pp. 45–48.
8. CISA. Lenovo Superfish Adware Vulnerable to HTTPS Spoofing. Available online: https://www.cisa.gov/news-events/alerts/2015/02/20/lenovo-superfish-adware-vulnerable-https-spoofing (accessed on 4 October 2024).
9. Hoogstraaten, H. Black Tulip: Report of the Investigation into the DigiNotar Certificate Authority Breach. 2012. Available online: https://www.researchgate.net/publication/269333601_Black_Tulip_Report_of_the_investigation_into_the_DigiNotar_Certificate_Authority_breach?channel=doi&linkId=5486fcf80cf268d28f06fa61&showFulltext=true (accessed on 4 October 2024).
10. Chowdhury, T.; Lesiuta, E.; Rikley, K.; Lin, C.-W.; Kang, E.; Kim, B.; Shiraishi, S.; Lawford, M.; Wassyng, A. *Safe and Secure Automotive Over-the-Air Updates*; Springer: Cham, Switzerland, 2018.
11. Fizza, K.; Auluck, N.; Azim, A.; Maruf, M.A.; Singh, A. Faster OTA Updates in Smart Vehicles Using Fog Computing. In Proceedings of the 12th IEEE/ACM International Conference on Utility and Cloud Computing Companion (UCC'19 Companion), New York, NY, USA, 2–5 December 2019; pp. 59–64. [CrossRef]
12. Kent, D.; Cheng, B.H.; Siegel, J. Assuring Vehicle Update Integrity Using Asymmetric Public Key Infrastructure (PKI) and Public Key Cryptography (PKC). In Proceedings of the IEEE International Conference on Vehicular Electronics and Safety, Cairo, Egypt, 4–6 September 2019; IEEE: Piscataway, NJ, USA, 2019.
13. Manna, M.L.; Treccozzi, L.; Perazzo, P.; Saponara, S.; Dini, G. Performance Evaluation of Attribute-Based Encryption in Automotive Embedded Platform for Secure Software Over-the-Air Update. *Sensors* **2021**, *21*, 515. [CrossRef] [PubMed]
14. Ghosal, A.; Halder, S.; Conti, M. STRIDE: Scalable and Secure Over-The-Air Software Update Scheme for Autonomous Vehicles. In Proceedings of the IEEE International Conference on Communications, Dublin, Ireland, 7–11 June 2020; IEEE: Piscataway, NJ, USA, 2020. [CrossRef]
15. Plappert, C.; Fuchs, A. Secure and Lightweight Over-the-Air Software Update Distribution for Connected Vehicles. In Proceedings of the 39th Annual Computer Security Applications Conference, Austin, TX, USA, 4–8 December 2023; pp. 268–282. [CrossRef]
16. Seo, J.W.; Islam, A.; Masuduzzaman, M.; Shin, S.Y. Blockchain-Based Secure Firmware Update Using an UAV. *Electronics* **2023**, *12*, 2189. [CrossRef]
17. Baza, M.; Nabil, M.; Lasla, N.; Fidan, K.; Mahmoud, M.; Abdallah, M. Blockchain-Based Firmware Update Scheme Tailored for Autonomous Vehicles. In Proceedings of the 2019 IEEE Wireless Communications and Networking Conference (WCNC), Marrakesh, Morocco, 15–18 April 2019; pp. 1–7.
18. Oham, C.; Michelin, R.; Kanhere, S.S.; Jurdak, R.; Jha, S. B-FERL: Blockchain-Based Framework for Securing Smart Vehicles. *arXiv* **2020**. [CrossRef]
19. Choi, S.; Lee, J.-H. Blockchain-Based Distributed Firmware Update Architecture for IoT Devices. *IEEE Access* **2017**, *8*, 37518–37525. [CrossRef]
20. Sobti, R.; Geetha, G. Cryptographic Hash Functions: A Review. *Int. J. Comput. Sci. Issues* **2012**, *9*, 461.
21. Rauchs, M.; Glidden, A.; Gordon, B.; Pieters, G.; Recanatini, M.; Rostand, F.; Vagneur, K.; Zhang, B.Z. Distributed Ledger Technology Systems: A Conceptual Framework. *SSRN Electron. J.* **2018**, 15. [CrossRef]

22. Werth, J.; Berenjestanaki, M.H.; Barzegar, H.R.; el Ioini, N.; Pahl, C. A Review of Blockchain Platforms Based on the Scalability, Security, and Decentralization Trilemma. In Proceedings of the International Conference on Enterprise Information Systems (ICEIS), Prague, Czech Republic, 24–26 April 2023; Volume 1, pp. 146–155.
23. Sealey, N.; Aijaz, A.; Holden, B. Iota Tangle 2.0: Toward a Scalable, Decentralized, Smart, and Autonomous IoT Ecosystem. In Proceedings of the 2022 International Conference on Smart Applications, Communications and Networking (SmartNets), Palapye, Botswana, 29 November–1 December 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–8. [CrossRef]
24. Fartitchou, M.; Lamaakal, I.; Maleh, Y.; El Makkaoui, K.; El Allali, Z.; Pławiak, P.; Alblehai, F.; Abd El-Latif, A.A. IOTASDN: IOTA 2.0 Smart Contracts for Securing Software-Defined Networking Ecosystem. *Sensors* **2024**, *24*, 5716. [CrossRef] [PubMed]
25. W3C. World Wide Web Consortium Recommendation. 2024. Available online: https://www.w3.org/ (accessed on 1 October 2024).
26. Mazzocca, C.; Acar, A.; Uluagac, S.; Montanari, R.; Bellavista, P.; Conti, M. A Survey on Decentralized Identifiers and Verifiable Credentials. *arXiv* **2024**. [CrossRef]
27. Khan, R.; McLaughlin, K.; Laverty, D.; Sezer, S. STRIDE-based threat modelling for cyber-physical systems. In Proceedings of the 2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), Turin, Italy, 26–29 September 2017. [CrossRef]
28. Mahmood, S.; Nguyen, H.N.; Shaikh, S.A. Systematic threat assessment and security testing of automotive over-the-air (OTA) updates. *Veh. Commun.* **2022**, *35*, 100468. [CrossRef]
29. *ISO/SAE 21434:2021*; Road Vehicles—Cybersecurity Engineering; International Organization for Standardization and Society of Automotive Engineers. 2021. Available online: https://www.iso.org/standard/70918.html (accessed on 19 November 2024).
30. IOTA Foundation. IOTA 2.0 AMA. IOTA Blog. Available online: https://blog.iota.org/iota-20-ama/ (accessed on 6 November 2024).
31. Tesla, Inc. Model 3 Owner's Manual. Available online: https://www.tesla.com/ownersmanual/model3/en_us/Owners_Manual.pdf (accessed on 6 November 2024).
32. Nie, S.; Liu, L.; Du, Y. Free-Fall: Hacking Tesla from Wireless to CAN Bus. In Proceedings of the Briefing, Black Hat USA, Las Vegus, NV, USA, 22–27 July 2017; Volume 25, p. 16.
33. Gligoric, N.; Escuín, D.; Polo, L.; Amditis, A.; Georgakopoulos, T.; Fraile, A. IOTA-Based Distributed Ledger in the Mining Industry: Efficiency, Sustainability and Transparency. *Sensors* **2024**, *24*, 923. [CrossRef] [PubMed]
34. Kahmann, F.; Honecker, F.; Dreyer, J.; Fischer, M.; Tönjes, R. Performance Comparison of Directed Acyclic Graph-Based Distributed Ledgers and Blockchain Platforms. *Computers* **2023**, *12*, 257. [CrossRef]
35. MOBI. MOBI Vehicle Identity Standard (VID) Version 2.1. 2024. Available online: https://dlt.mobi/wp-content/uploads/2024/03/MOBI-VID0001WP2021-Version-2.1.pdf (accessed on 29 September 2024).
36. Sun, X.; Yu, F.R.; Zhang, P.; Sun, Z.; Xie, W.; Peng, X. A Survey on Zero-Knowledge Proof in Blockchain. *IEEE Netw.* **2021**, *35*, 198–205. [CrossRef]
37. Belchior, R.; Scuri, S.; Nunes, N.; Hardjono, T.; Vasconcelos, A. Towards a Standard Framework for Blockchain Interoperability: A Position Paper. *preprints* **2024**.
38. IOTA Foundation. LayerZero Integrates with IOTA EVM. Available online: https://blog.iota.org/layerzero-integrates-with-iota-evm (accessed on 3 October 2024).
39. Rodriguez Garzon, S.; Natusch, D.; Philipp, A.; Küpper, A.; Einsiedler, H.J.; Schneider, D. DID Link: Authentication in TLS with Decentralized Identifiers and Verifiable Credentials. *arXiv* **2024**. [CrossRef]
40. Yu, G.; Wang, X.; Yu, K.; Ni, W.; Zhang, J.A.; Liu, R.P. Survey: Sharding in Blockchains. *IEEE Access* **2020**, *8*, 14155–14181. [CrossRef]