Efficiently Verifying IoT State in Web3: A Succinct Proof Approach for Scalable and Transparent Applications

Jacob Everly
College of Engineering
Cornell Tech
New York, New York
Email: je354@cornell.edu

Hau Chu
College of Engineering
Cornell Tech
New York, New York
Email: hc793@cornell.edu

***Abstract*—The rapid expansion of Internet of Things (IoT) devices necessitates the development of secure, efficient, and interoperable solutions to integrate them into Web3 systems. Motivated by the need to enable IoT adoption in Web3, this research paper proposes a novel method for constructing optimized virtual machines (VMs) for IoT devices' data packages using a combination of Compact Certificates (CCs) and Succinct proofs. By leveraging CCs and Succinct proofs, we can effectively reduce the complexity of state verification, allowing for efficient cross-chain interaction and state proof export. Assuming a background in blockchain, this paper presents the system architecture comprising blockchain, CC creation, CC verification, and state proof export. Our approach demonstrates the potential of CCs and Succinct proofs for building interoperable blockchain networks, significantly improving the security and efficiency of IoT integration in existing systems while fostering user trust and transparency.**

I. Introduction

The Internet of Things (IoT) has the potential to revolutionize various aspects of our daily lives by integrating billions of interconnected devices into diverse ecosystems. Some real-world examples include wearables, smart homes, manufacturing sensors, and emission monitoring systems. However, the successful integration of IoT devices into multi-chain systems necessitates addressing critical challenges like security, privacy, and interoperability. Blockchain technology has emerged as a promising solution, but the fragmentation of blockchain networks hinders its full potential.

This paper investigates the use of Compact Certificates (CCs) as a means to enhance interoperability across multi-chain mechanisms and facilitate the adoption of IoT devices in the Web3 space. Although CCs are a well-established technique within the blockchain community, their application in IoT systems has yet to be extensively studied. By utilizing CCs, we can develop optimized virtual machines (VMs) and state proof protocols for IoT chains, effectively addressing secure state verification challenges in multi-chain environments.

Our research delves into the design and implementation of CCs, demonstrating their ability to streamline state verification across different blockchains. This efficiency reduces the time and costs associated with cross-chain interactions, creating a more secure and efficient infrastructure for IoT devices. It is essential to note that our work concentrates on applying CCs to build interoperable blockchain networks for IoT devices, rather than on the development of CCs themselves.

In summary, this paper underscores the potential of CCs in tackling the challenges of integrating IoT devices into existing systems while improving the efficiency and security of IoT chains. By offering insights into the practical benefits of CCs for constructing interoperable blockchain networks, we contribute to the advancement of more secure and efficient IoT systems that necessitate user trust and transparency.

A. Related Work

The concept of Compact Certificates (CCs) was first introduced in[1]. They proposed compact certificate schemes as a method for compressing a large number of signatures on a message M from signers with varying weights into a significantly shorter certificate. This certificate enables verifiers to be convinced that signers with sufficient total weight signed M without having to see or verify all signatures. As a practical application, compact certificates can be used to prove that parties with a sufficient total account balance have attested to a given block in a blockchain.

Our research builds upon this foundation by focusing on the application of CCs within blockchain networks and IoT devices to enhance interoperability across multi-chain mechanisms. A noteworthy implementation of CCs is observed in the Algorand blockchain, which leverages CCs to generate state proofs, facilitating the efficient export of their state and bolstering the post-quantum security of their consensus protocol. Our paper delves into the design and implementation of CCs, showcasing their efficacy. The insights derived from this research contribute to the development of secure and efficient IoT systems that require user trust and transparency.

B. Emissions Network Application

Emission networks are systems designed to monitor and report environmental data, such as greenhouse gas emissions and air quality, through a vast array of sensors. Ensuring the

trustworthiness and transparency of this data is crucial for stakeholders affected by emissions, regulators, and decision-makers. Blockchain technology can provide the necessary transparency and security for emission data, thereby promoting trust and accurate reporting.

Compact Certificates (CCs) can enhance the management of emission networks by facilitating transparent storage and transmission of data from these networks to end-users. Our paper proposes the use of CCs to relay data from custom emission chains, featuring optimized virtual machines (VMs) for IoT sensors, to provide a secure and efficient method for storing and transmitting this data to end-users. With CCs, the organization managing the data can ensure that the data is transmitted and stored transparently, while end-users can verify the authenticity and validity of the data using the attester tree and the Succinct Proof generated at the end.

CCs enable verification with larger attester sizes, allowing more stakeholders to participate in the signing process which allows all of the stakeholders in an emmision network to participate in signing the CC. This allows end-users to act on the data with confidence, fostering greater transparency and trust in the management of emission data. Our research demonstrates the practical application of CCs in the management of emission data and highlights their potential to address the challenges facing the integration of IoT devices into existing systems.

II. Background

To fully appreciate the application of Compact Certificates (CCs) in the context of blockchain networks and IoT devices, it is essential to have a solid understanding of the fundamental technologies and concepts that underpin them. This section offers a concise overview of crucial concepts related to basic blockchain infrastructure ideas such as Merkle trees, light and full nodes, compact certificates, slasher algorithms, signature schemes, and succinct proofs. While our target audience is expected to have prior knowledge of these concepts, we provide a high-level review to ensure a shared understanding. Furthermore, familiarity with the Compact Certificate paper by Micali et al. and Algorand's implementation of the concept will be beneficial in grasping our proposed methodology.

A. Merkle Trees

Merkle trees are a binary tree data structure commonly used in blockchain technology for efficient storage and verification of large datasets. The structure allows for the creation of a root hash that represents the entire dataset, enabling fast verification of data integrity without transferring or storing the entire dataset. In the context of blockchain networks, Merkle trees are utilized to store transaction data, enabling efficient and secure state verification across different chains. Additionally, Merkle trees are critical in the creation of Compact Certificates by serving as the basis for the attester tree and ground truth, which are utilized to represent and verify the state of a chain in an efficient manner. We provide a detailed explanation of attester trees and ground truth later in the paper.

B. Light vs Full nodes

In blockchain networks, nodes can be classified as either full nodes or light nodes. Full nodes are nodes that download and store the entire blockchain, enabling them to validate transactions and blocks independently. Full nodes are computationally expensive to operate due to their high storage and processing requirements. Light nodes, on the other hand, do not download and store the entire blockchain but rely on full nodes for transaction validation. Light nodes are less computationally expensive to operate and require less storage space than full nodes. However, light nodes sacrifice some level of security and privacy since they must trust full nodes for validation. In the context of IoT devices, light nodes may be more suitable due to their lower computational and storage requirements, making them a more feasible option for resource-limited devices.

C. EdDSA signature scheme

EdDSA is a digital signature algorithm based on the elliptic curve Ed25519, designed to provide security, speed, and efficiency for use in blockchain networks. Utilizing elliptic curve cryptography, it generates and verifies digital signatures with small key sizes, making it ideal for resource-limited environments like IoT devices.

D. Compact Certificates

The Compact Certificate paper by Micali et al. (2018) introduces a novel cryptographic primitive called Compact Certificates, which enable the efficient compression and verification of multiple signatures on a message by signers with different weights [1]. The paper demonstrates an efficient compact certificate scheme and its implementation in a decentralized setting over an unreliable network, even in the presence of adversarial parties who wish to disrupt certificate creation. The evaluation shows that compact certificates are significantly smaller and cheaper to verify than a natural baseline approach. We highly recommend reading the original paper to fully understand the concept of Compact Certificates. Algorand has adopted Compact Certificates, which has helped to advance their protocol by making their consensus post-quantum secure.

E. Slasher Algorithm

Slasher algorithms are a class of Proof-of-Stake (PoS) algorithms that address some of the security and scalability issues of PoS systems, as introduced by Vitalik Buterin in his 2014 paper [2]. In PoS systems, validators are chosen based on their stake and are responsible for creating and validating new blocks in the blockchain. Slasher algorithms improve the security of PoS systems by penalizing validators who create multiple blocks with the same stake, a practice known as "double-signing." Validators who engage in such practices are penalized by reducing their stake, serving as an incentive for validators to behave honestly. Slasher algorithms have been implemented in several PoS-based blockchain networks, including Ethereum, which has successfully improved their security and scalability.

F. Succinct Proofs

Succinct proofs are cryptographic proofs that enable the validation of a piece of information using significantly less storage space than traditional proofs. They were first introduced by Eli Ben-Sasson et al. in their 2013 paper [3]. Succinct proofs enable efficient verification of complex computations or statements by using specialized data structures and cryptographic techniques to compress large amounts of information into a small, easily verifiable proof. They have wide-ranging applications in areas such as blockchain networks, digital identity, and data privacy, where efficient and secure storage and validation of information are critical. Succinct proofs have the potential to greatly improve the efficiency and

G. Most Significant Bit vs Least Significant Bit

The Most Significant Bit (MSB) and Least Significant Bit (LSB) are the two ends of a binary sequence, with the MSB being the leftmost bit and the LSB being the rightmost bit. In the context of Merkle trees, the use of MSB or LSB can impact the order in which leaves are hashed, ultimately affecting the root hash of the tree. The choice between MSB and LSB can affect the performance and security of the system. We choose to have LSB in our project and provide a detailed explanation for this decision later in the paper. MSB-first traversal does not offer any additional security benefits compared to LSB-first traversal.

III. Defining our Web3 IoT systems

To provide a comprehensive understanding of how we implement our proposed Web3 IoT system for emission data management, this section delves into the specifics of the system's components and their roles. These components include emission sensors acting as lite clients, validators and attesters, the chosen consensus mechanism, and relay contracts for cross-chain communication. This detailed examination of the system components is crucial for grasping the practical applications

A. Emission Sensors as Lite Clients

In our proposed Web3 IoT system, the pollutant sensors will be implemented as lightweight clients responsible for reporting emission data to the chain. The reliability of emission data reported by these sensors is crucial for the effectiveness of our system. To ensure the accuracy of the data, each sensor must be properly calibrated and comply with recognized data reporting standards such as ISO 14064 before being allowed to report. These measures help to increase the reliability and validity of the data being reported, enabling our system to provide a transparent and trustworthy platform for the management of emission data. By leveraging these calibrated sensors, our proposed Web3 IoT system aims to provide a practical and efficient solution for the management of emission data.

B. Validator and Attester Set

Validation and attestation are critical components of ensuring the accuracy and reliability of the pollutant sensor data in our Web3 IoT system. Our system relies on a network of validators who also act as attesters for creating Compact Certificates (CCs) that represent the state of the chain. Validators must fulfill specific requirements to run a validating node on the chain, including running an emission sensor, participating in creating CCs, and staking a native coin. The weight of each validator's vote is proportional to the percentage of staked coins they hold compared to the total number of tokens staked on our platform. This approach incentivizes participants to contribute more to the network and ensures that their vote carries appropriate weight in the decision-making process. Validators who fail to vote or validate accurately may face the slashing of staked coins, further encouraging responsible behavior and participation in the network. The involvement of validators and attesters is crucial for the proper functioning of our Web3 IoT system, as it ensures the accurate reflection of data reported by pollutant sensors on the chain.

C. Consensus Mechanism

To achieve consensus on the chain, our validators will be running a Proof of Stake (PoS) consensus mechanism. This ensures that validators who are staking more coins on the platform have a higher chance of being selected to validate and create the next block in the chain. PoS mechanisms have been shown to be more energy-efficient and cost-effective than other consensus algorithms, making it a suitable choice for our Web3 IoT system. Additionally, PoS mechanisms have demonstrated their security in systems such as Ethereum and enable us to determine the weight of an attester's vote. Validators must continuously monitor the chain for any irregularities and act quickly to resolve any issues that may arise to maintain the integrity of the system. We assume that our system inherits all of the security guarantees of a traditional PoS system, including resistance to 51 percent attacks and other common vulnerabilities. To ensure efficient attestation of the current agreed state of the chain, our data structure will be sequential, avoiding some complications that arise when attesters want to sign a message on a specific state of the chain.

D. Relay Contracts

Relay contracts are a type of smart contract that enables communication and data transfer between different blockchain networks. These contracts act as intermediaries between two chains, allowing for events to be triggered on one chain and then relayed to another chain. This is accomplished by creating an event listener on one chain that listens for a specific event to occur, and then transmitting the relevant data to a relay contract. The relay contract then sends the data to the target chain, where it can be processed and executed accordingly.

IV. System Architecture

In this section, we outline the architecture of our proposed Web3 IoT system and its various components, as depicted in

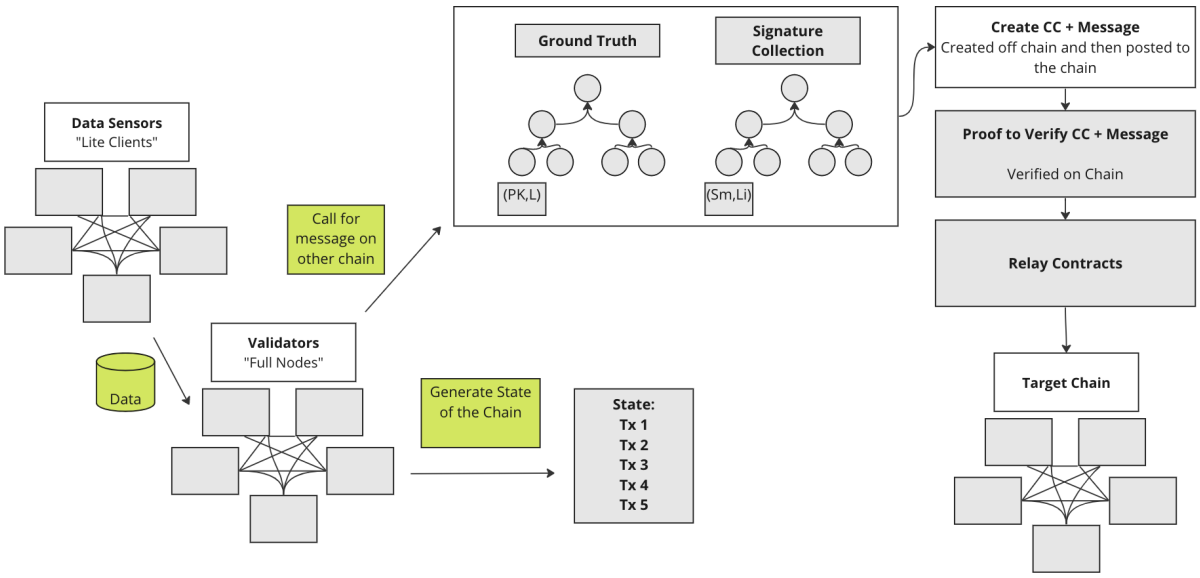

Figure 1. System Overview

Figure 1. These components include compiling data on the chain, maintaining the state of the system, collecting signatures for the certificate, creating and verifying compact certificates, and exporting them to enable cross-chain interactions. The following subsections provide a detailed description of each process and its role in the overall system.

A. Compiling Data on Chain

Our system uses data sensors, specifically emission sensors, to collect and report data to the blockchain. The blockchain employs an optimized Virtual Machine (VM) tailored for IoT devices, allowing them to process and store their data packages efficiently. The VM minimizes resource consumption while ensuring data reliability and accuracy. This process can be seen on the left side of Figure 1.

B. Maintaining the State of the system

To maintain the system's state, nodes running emission sensors act as validators. Nodes must prove they are running a sensor correctly and stake a predetermined amount of cryptocurrency. This stake serves as the validator's weight and contributes to the creation of the Compact Certificate. The staking mechanism ensures the proper functioning of the network and enhances its security. This process is illustrated in the bottom left of Figure 1.

C. Collecting Signatures for Certificate

The process of collecting signatures and signed messages for the Compact Certificate begins with a user requesting it. First, we gather all public keys and weights of the attestors, creating a Merkle tree to serve as an index of valid voters. Validators then submit signed messages, each assigned a weight range determined by their staked amount. Once the threshold of stake is reached, the voting process is halted. Two separate Merkle trees are subsequently created: one containing the public keys and weights of all potential voters and another comprising signed messages and corresponding attester weight ranges. This process is depicted on the top of Figure 1 and shown in a close-up in Figure 2.

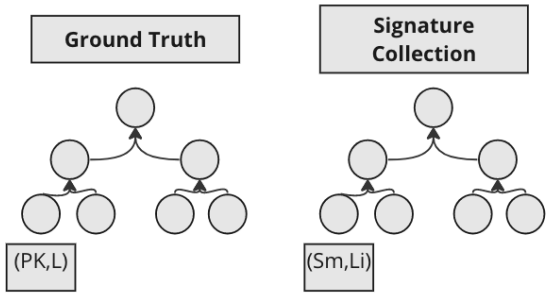

Figure 2. Data Needed for Compact Certificate. PK is Public Key, L is the attesters weight, Sm is the signed messaged and Li is the weight range of the attester

D. Creating Compact Certificates

To create the compact certificate, we adhere to the steps outlined in Silvio Micali's whitepaper, "Compact Certificates of Collective Knowledge"[1]. This process is carried out using off-chain computation, and the resulting compact certificate is then posted on the blockchain. This process is represented in the top right of Figure 1.

E. Verifying Compact Certificate

Verification of the compact certificate is achieved using the validity tests from Micali's paper[1]. The user responsible for creating the certificate posts it on the blockchain. Then, the

verification process occurs on-chain, ensuring that the compact certificate is authentic and reliable before it is accepted by the network. This on-chain verification enhances the security and trustworthiness of the system. This process is displayed on the right side of Figure 1.

F. Exporting Compact Certificate

A smart contract residing on the home chain handles the storage of the compact certificate and its verification proof. To facilitate cross-chain interactions, a relay contract can be employed to notify another chain's contract or user that the proof has been completed. Users on the recipient chain can then easily check the proof's validity, ensuring the integrity of the information. This process is illustrated in the bottom right of Figure 1.

V. Security Considerations

In this section, we discuss the security considerations of our proposed Web3 IoT system, focusing on the strategies we employ to ensure the integrity and robustness of the network. We explore the use of the least significant bit in Merkle tree construction, the implementation of a Slasher algorithm to ensure the proper broadcasting of signatures, and mechanisms for preventing equivocation from validators. These measures aim to safeguard the system against malicious actors and foster a secure environment for data management and processing.

A. Using Least Significant Bit for Merkle Tree Construction

To address security concerns, we employ the least significant bit (LSB) when constructing the Merkle tree. Using LSB eliminates ambiguity that could be exploited by a malicious actor lying about the tree's size. However, this method interferes with our optimization of condensing Merkle tree paths by placing higher weights on one side of the tree. To overcome this issue, we precompute LSB positions and permute the higher-weighted signatures to one side again. This approach is based on the argument that LSB-first numbering removes ambiguity regarding which side of the tree to traverse, independent of the bit string's length. This choice aligns with our earlier discussion on selecting LSB for our project.

B. Insuring Signatures Broadcasts

To mitigate the risk of Denial of Service (DoS) attacks, we implement a Slasher algorithm for collecting signed messages during certificate creation. Nodes are required to participate in the submission of signed messages and can monitor the behavior of other nodes. Those failing to act appropriately will be penalized, similar to the Slasher algorithm used in Ethereum, as proposed in[4]. This approach is consistent with our previous mention of the Slasher algorithm.

C. Preventing Equivocation from Validators

To discourage validators from misbehaving, our system requires both capital and time investments. First, users must provide proof that their sensor is operating correctly and has incurred a significant capital investment. Next, validators must stake cryptocurrency. Any misbehavior results in the loss of their validation privileges and staked coins, effectively forfeiting their investments in the sensor and cryptocurrency. This mechanism deters validators from acting maliciously and promotes honest behavior within the network.

VI. Conclusion

In this paper, we have explored the potential of Compact Certificates (CCs) and Succinct proofs as powerful tools to foster IoT adoption in Web3 systems by enabling the construction of optimized virtual machines (VMs) for IoT devices' data packages. By effectively reducing the complexity of state verification, we demonstrated how CCs and Succinct proofs facilitate efficient cross-chain interaction and state proof export, addressing the security and efficiency challenges inherent in IoT integration.

Throughout our research, we have delved into the system architecture, detailing the components of blockchain, CC creation, CC verification, and state proof export. Our work highlights the practical benefits of CCs and Succinct proofs in building interoperable blockchain networks, which in turn contribute to the development of more secure, efficient, and transparent systems for IoT devices.

In conclusion, this paper emphasizes the transformative impact of combining CCs and Succinct proofs on the integration of IoT devices into existing systems. Our approach holds the promise of revolutionizing IoT adoption in Web3 systems, significantly improving security, efficiency, and user trust. By offering insights into these innovative technologies, we hope to drive further research and development in the pursuit of creating a seamless, interoperable, and transparent future for IoT devices within the realm of Web3.

A. Limitations and Future Work

It is crucial to acknowledge the limitations and potential future work in our proposed methodology employing Compact Certificates (CCs) and Succinct proofs. While our approach effectively addresses the challenges of IoT integration in Web3 systems, it does not directly focus on optimizing virtual machines (VMs) for IoT sensors or the accuracy of the data relayed from the sensors themselves.

Future work in this area could concentrate on refining VMs for IoT sensors and devising more robust algorithms for data validation, ensuring the reliability and precision of transmitted data. Moreover, the efficiency and security of CCs and Succinct proofs can be further enhanced by exploring alternative data structures and cryptographic techniques.

Pursuing research in these directions holds the potential to address the current limitations of our methodology and broaden its practical applications, not only in the context of emission data management but also in various other domains. As IoT devices continue to proliferate, the development and refinement of secure, efficient, and interoperable solutions will be instrumental in realizing the full potential of Web3 systems.

ACKNOWLEDGMENT

We would like to express our deepest gratitude to our professor, Anna Scaglione, for her invaluable guidance, support, and mentorship throughout this research project. Her expertise, constructive criticism, and encouragement have been instrumental in shaping our work and allowing us to explore this fascinating topic.

Additionally, we would like to extend our sincere thanks to Riad Wahby, who introduced us to the subject of Compact Certificates and provided crucial insights and direction to ensure our correct implementation throughout the project. His knowledge and assistance have been vital in our understanding and development of this research.

REFERENCES

[1] Silvio Micali, Leonid Reyzin, Georgios Vlachos, Riad S. Wahby, and Nickolai Zeldovich. Compact certificates of collective knowledge. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 626–641, 2021.

[2] Vitalik Buterin. Ethereum white paper: A next generation smart contract & decentralized application platform. 2013.

[3] Eli Ben-Sasson, Alessandro Chiesa, Eran Tromer, and Madars Virza. Succinct Non-Interactive zero knowledge for a von neumann architecture. In *23rd USENIX Security Symposium (USENIX Security 14)*, pages 781–796, San Diego, CA, August 2014. USENIX Association.

[4] Vitalik Buterin. Slasher: A punitive proof-of-stake algorithm, 2014.