# Technische Universität Berlin

Information Systems Engineering
Fakultät IV
Einsteinufer 17
10587 Berlin
https://www.tu.berlin/ise

Bachelor's Thesis

# Enabling Privacy-Preserving Policy Compliance in Local Renewable Energy Markets

Stefan Valentino Krakan

Matriculation Number: 450838

27.03.2024

Supervised by
Prof. Dr. Stefan Tai
Prof. Dr. Axel Küpper

Assistant Supervisors
Dr.-Ing. Sebastian Werner
Alvaro Alonso Domenech

Hereby I declare that I wrote this thesis myself with the help of no more than the mentioned literature and auxiliary means.

Berlin, 27.03.2024

Stefan Krakan

Signer ID: MCL0PRXI12...

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

*(Signature Stefan Valentino Krakan)*

## Abstract

As the world is targeting climate change, the need for households to transition to sustainable energy has never been more important. While each household's energy consumption might seem minor in isolation, collectively, they form a substantial part of the overall energy consumption. In this context, an aggregated view of collective renewable energy compliance across households becomes essential to effectively monitor the success of the renewable energy transition and environmental policies. However, monitoring this transition presents challenges, as revealing energy data poses severe risks to privacy, while on the other hand privacy poses the potential of being misused for deviant behaviour.

Therefore, this thesis addresses the paradoxical challenge of achieving transparency in the compliance process while preserving privacy in local renewable energy markets. It achieves this by employing Zero-Knowledge Proofs, which are structured in a nested architecture to allow for collective and layered validations without compromising privacy. Furthermore, a proof of concept leveraging the ZoKrates framework is implemented to demonstrate how aggregated energy data from various households can be collectively assessed against a predefined energy threshold. This compliance assessment helps to verify whether communities are meeting their renewable energy consumption targets. Lastly, the proof of concept is evaluated on its scalability, data management capabilities and practical applicability.

## Zusammenfassung

Um den Klimawandel effektiv anzugehen, ist der Übergang von Haushalten zu erneuerbaren Energien wichtiger denn je. Auch wenn der Energieverbrauch eines einzelnen Haushalts gering erscheinen mag, tragen alle Haushalte zusammen betrachtet einen erheblichen Anteil zum gesamten Energieverbrauch bei. Zusätzlich ist es wichtig, einen umfassenden Einblick in die kollektive Compliance von Haushalten zu erhalten, um festzustellen, ob die Energiewende und andere Umweltinitiativen wirksam sind. Allerdings gestaltet sich die Überprüfung dieser Compliance als herausfordernd. Einerseits birgt die Offenlegung von Energiedaten erhebliche Risiken für die Privatsphäre. Andererseits besteht die Gefahr, dass durch den Schutz der Privatsphäre betrügerisches Verhalten erleichtert wird.

Die vorliegende Arbeit behandelt diese paradoxe Herausforderung, kollektive Compliance-Transparenz in lokalen und erneuerbaren Energiemärkten zu erreichen, ohne dabei die Privatsphäre zu beinträchtigen. Dies wird mit Hilfe von Zero-Knowledge Proofs erreicht, die in einer verschachtelten Architektur angeordnet sind, um kollektive Validierungen bei gleichzeitigem Schutz der Privatsphäre zu ermöglichen. Darüber hinaus wird ein Proof-of-Concept mithilfe des ZoKrates-Toolkits entwickelt, das veranschaulicht, wie aggregierte Energiedaten aus verschiedenen Haushalten kollektiv gegen einen festgelegten Energie-Schwellenwert überprüft werden können. Solch ein System erlaubt es zu überprüfen, ob Gemeinden ihre Ziele bezüglich der Nutzung erneuerbarer Energien erreichen. Abschließend wird das Proof-of-Concept auf seine Skalierbarkeit, Datenmengenverwaltung und praktische Anwendbarkeit hin evaluiert.

# Contents

# List of Figures

# List of Tables

# 1 Introduction

In recent years climate change has become one of the most important issues for humanity. To target climate change effectively, carbon emissions need to be reduced and the transition towards sustainable energy has to be achieved. This global crisis requires action on multiple fronts, such as industry, transportation and living. While each of these sectors are equally important to transition to renewable energy sources in order to lower carbon emissions, this thesis will focus on individual households for the purpose of simplicity and demonstrability. Nevertheless, all ideas and concepts introduced in this thesis can be applied to sectors beyond households.

The significance of focusing on individual households in the context of energy consumption and climate change becomes clear when considering recent data. In 2021, the German Federal Environmental Agency (Umweltbundesamt) [1] reported that private households in Germany accounted for approximately 25 percent of the country's total energy consumption. While each household's energy consumption might seem minor in isolation, collectively, they form a substantial part of the overall energy footprint. It becomes therefore crucial to transition the majority of households to renewable energy, in order to target climate change effectively.

One critical aspect of this transition is the effective monitoring of the transition to renewable energy sources in households. This provides a tangible way to assess the impact and effectiveness of climate policies and environmental initiatives. Furthermore it enables the tracking of progress and provision of incentives, while also holding households accountable for their energy consumption and environmental impact.

As our society becomes increasingly digitised, so too do households by adopting smart metering technology that measures energy consumption and production in real time. The resulting data is mostly utilised to improve the billing process or improve the matching of supply and demand of energy. However, utilising the energy data, for example renewable energy production data of households, to monitor the implementation and impact of policies is challenging and requires a number of considerations.

First of all, it is essential to accurately measure energy consumption and production of households to provide meaningful data for compliance checks. Aggregating readings from individual households becomes necessary to get a comprehensive view of collective renewable energy consumption across households. The issue is, that energy data within these local renewable energy markets needs to be transparent enough to facilitate accurate and tamper-proof compliance checks. Without transparency, trustworthiness is not guaranteed, since entities can lie about their energy usage or falsely claim compliance. This potentially undermines the effectiveness of environmental policies. Yet, this transparency presents a risk due to the sensitive nature of energy data. Full transparency for integrity assurance conflicts with risks to privacy and data security of individual households. Even with a small-scale deployment of smart metering technology, it is possible to deduce information such as household occupancy, eating routines or even the religion of residents [2]. In the wrong hands, this data can be misused to harm individuals.

This issue can be further complicated by the global increase in state monitoring and data collection in recent years, as Eck et al. [3] note. They caution that such practices could pose threats to privacy rights and democratic norms, which emphasises the need for systems that can fulfil societal goals without compromising these fundamental principles.

Moreover, the trends towards environmental regulation, compliance and laws are increasing, as the 38-fold increase in environmental protection laws from 1972 to 2019 indicates. Even though the UN report [4] acknowledges that these environmental laws often lack enforcement, it advocates for an increase in monitoring and enforcing these regulations and laws.

However, building systems that achieve societal objectives, such as monitoring the transition to renewable energy sources, without sacrificing privacy seems to be difficult. As previously discussed, the need for transparency in the compliance process seems to be unachievable alongside privacy. Yet, the aim of this thesis is to develop a system that allows effective and verifiable compliance checks at a collective level, while still protecting individual privacy, as can be seen in Figure 1.1. In order to accomplish this, the *nesting* of Zero-Knowledge Proofs (ZKP) will be applied, which will be explained in detail in the following chapter, to create privacy-preserving, verifiable and collective compliance checks for local renewable energy markets.

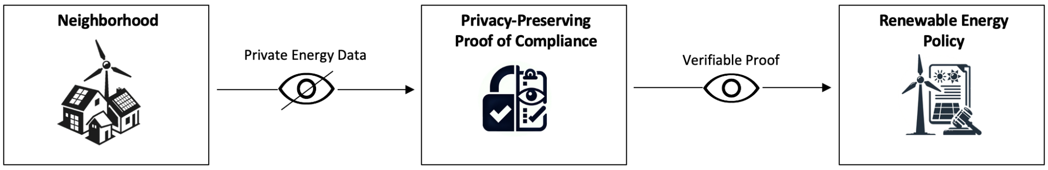

Figure 1.1: Privacy-preserving verifiable proof of compliance for renewable energy consumption

Building systems, that achieve this balance between compliance and privacy might be a key factor in gaining public trust and support for climate action policies. It might be a crucial area of study in targeting climate change. Furthermore, such systems can reduce state intervention and monitoring, while simultaneously empowering communities by encouraging shared responsibilities and collective-driven initiatives. Additionally by showing that privacy-preserving compliance works for this particular use case, this thesis could serve as a point of reference for larger-scale implementations or similar ideas, thereby setting a precedent.

## 1.1 Outline

We will start by exploring the background and fundamentals necessary to understand the concepts introduced in this thesis. Afterwards a review of existing literature regarding similar problems and approaches is conducted, which will provide a foundation for the formulation of this thesis' research question in chapter four. Following this, the proof of concept is introduced in four distinct chapters: starting with a more abstract view of the socio-economic consequences of such a system, followed by a chapter about the system design with its requirements and architectural design, on which a chapter about the actual implementation with various components will be build upon and finishing with an evaluation of the implementation. Lastly, the conclusion chapter will end this thesis.

# 2 Background

This chapter explains the importance of privacy and if compliance is even compatible with privacy. This foundation is essential to realise how ZKPs and their nesting play a crucial role in this thesis' implementation, where privacy-preservation is paramount. Further key concepts and relevant terms related to these are introduced as well.

## 2.1 Privacy-Preserving Compliance - A Paradox?

It appears that privacy and compliance contradict each other, as the former seeks to conceal information while the latter tries to reveal it. In this section I will establish my understanding of privacy as well as compliance and discuss if they truly contradict each other.

Figure 2.1: An iconographic representation of the intersection of privacy and compliance

### 2.1.1 Defining Privacy

Discussing all aspects of privacy is beyond the scope of this thesis, as there exists no universally accepted definition. Furthermore privacy is viewed differently across various disciplines like sociology, psychology, law and information technology. Nevertheless we aim to briefly introduce our understanding of privacy and highlight its key aspects, as it has become one of the most crucial social issues of the information age in the 21st century.

Etymologically, the term privacy originates from the Latin word *privatus*, which means set apart from what is "public, personal and belonging to oneself and not to the state" [5]. Historically, this primarily referred to property rather than information as it is understood today.

Alibeigi et al. [6] illustrate how the understanding of privacy has evolved over the centuries, from "a right to be let alone" in the 19th century to "a right to personal data protection" in recent times. They conclude that the right to privacy is primarily aimed at protecting "individual dignity, honour and esteem", by protecting personal possessions, such as property or data, from unpermitted access. Strickland et al. [7] note that the concept of privacy has recently broadened to include also "freedom from business intrusions". Privacy serves as a defence against "unwarranted publicity" and the potential "tarnishing of fundamental aspects of human personality" [6]. Compromising these aspects can lead to serious and irreparable damage.

According to Mustafa et al. [8], privacy preservation in local energy markets should protect the consumer's privacy to the greatest extent possible. Therefore, their *principle of least privilege* should be applied, which means entities should have access only to data necessary to complete their task. To illustrate this, they make the example, that there is in most cases no need to reveal the consumer's identities in energy markets.

In conclusion, privacy is a broad field with a seemingly simple yet complex concept to define. To clarify, I define privacy, similar to Alibeigi et al. [6], as an individual's "natural desire to be free from others' control and surveillance". Furthermore, it is the individual's right to choose how personal information is collected, used and shared. Specifically in local energy markets, the amount of disclosed data should be minimised as much as possible.

### The Risk to Privacy from Smart Metering

Especially smart metering is not without risks when considering its privacy implications. Having all energy data monitored in real-time and digitally available poses severe risks to privacy. Even with a small-scale deployment of smart metering technology, it is possible to reveal information like household occupancy, eating routines or even the religion of residents [2]. In the wrong hands, this data can be misused to harm individuals. This misuse can also raise concerns about potential surveillance by governments or companies. In countries with restrictive governments, or those that might become so, the detailed data collected by smart meters could potentially be used in discriminatory ways, for example for discrimination or prosecution of people of certain religions. Moreover, in most cases, individuals are unaware of the information that can be extracted about them through information technology, nor do they understand the potential consequences of such data [9]. This consideration is particularly relevant to smart metering technology. Not without reason does a report by the consumer union in the Netherlands [10] argue, that frequent smart meter readings may violate article 8 on Privacy of the European Convention on Human Rights and raises thereby questions about their legality. Additionally, they emphasise the need to balance smart metering benefits with potential privacy infringements. Garcia et al. [11] suggest that aggregated energy data on a neighbourhood level is often sufficient for grid operators or, as in this thesis, compliance auditors. They also highlight that the more frequently smart meter data is measured, the higher the risk of privacy violations. Therefore, they recommend using the least frequent measurements possible for each use case to mitigate these risks.

### 2.1.2 Defining Compliance

Compliance refers to the act of conforming to established guidelines, specifications or the process of ensuring that legal requirements, standards and regulations are met [12]. In organisations and businesses, compliance usually involves adhering to laws, industry standards and policies, for instance financial regulations, labour laws, environmental standards or data protection laws. Usually, compliance is only effective when validated by an independent authority, such as a regulatory agency. The mere existence of an authority with an absence of direct monetary benefits or explicit threats of sanctions leads to an increased compliant behaviour [13]. Moreover, this external oversight ensures that the compliance is not self-regulated or even fake, but objectively verified. Such an approval is necessary to ensure the integrity and legitimacy of the compliance process. Nonetheless, making compliance data accessible to auditors infringes upon privacy, as it discerns personal data, like financial information or smart meter readings.

However, intricacies arise when considering the determinism in assessing compliance. In this context, determinism would infer that the outcome of the compliance process is always consistent and predictable. Yet in practice it can be elusive, especially in dynamic and volatile environments like renewable energy markets. Factors like fluctuating energy production and consumption as well as evolving regulatory requirements introduce uncertainties that render the compliance process non-deterministic. Furthermore, this requires that the system functions always reliably and predictable, even in scenarios with data loss or other adverse conditions. This might be overly optimistic, because realistic scenarios are imperfect and uncertain, which must be accounted for in any compliance framework or system. Similarly, in the healthcare sector, Kyngäs et al. [14] highlight the lack of reliable and valid measures for compliance, along with the diversity of conditions and treatments, that make the operationalisation and measurement of compliance challenging. Consequently, incorporating a dynamic and updatable system of rewards and punishments into such frameworks becomes essential when faced with the non-deterministic nature of compliance, allowing the system to remain effective and fair under varying and unpredictable circumstances. In light of these considerations, terms like *conformance* or *adherence* may reflect more accurately the process of aligning with established rules and standards in this thesis, without overemphasising the deterministic and absolute connotations associated with compliance.

However, the term compliance will still be used in this thesis to denote the process of meeting specified criteria within a given framework, while recognising that this simplification is for clarity and conceptual focus. In this way, this thesis aims to cross the theoretical ideals of compliance with the practical realities of embedded systems and renewable energy markets.

### 2.1.3 Does Verifying Compliance Contradict Privacy?

At first glance, privacy and compliance seem to have an antagonistic relationship. The essence of this perceived contradiction lies in the traditional approach to compliance, as its verification requires access to sensitive data in order to ensure integrity and effectiveness of the assessment. However, this conflicts with privacy, as it requires organisations or individuals to disclose certain information that they might prefer to keep private, such as financial information or smart meter data.

Therefore, this thesis demonstrates that determining mere adherence or non-adherence to compliance standards, can respect privacy using technologies that protect sensitive data. As the focus lies on this binary end result, the preceding details are not of interest, provided that their integrity and correctness are ensured.

## 2.2 Achieving Privacy-Preserving Collective Compliance - Technological Fundamentals

As we have seen in the previous section, we need technology capable of producing such binary outcomes of compliant or non-compliant without disclosing any further information. This leads us to ZKPs. In the following section, ZKPs are introduced and explained, followed by an exploration of their more advanced concepts that enable the privacy-preserving collective compliance verification in this thesis proof of concept.

### 2.2.1 Zero-Knowledge Proofs

Initially introduced in 1985 [15], ZKPs are cryptographic protocols, building upon elliptic curve cryptography, that "enable a prover to confirm the validity of a statement to a verifier" without disclosing any additional information. The statement being proven is generally denoted as a mathematical assertion, for instance, "I know the value $x$ such that $f(x) = y$", where $f$ is a known function and $y$ is a known value.

In non-interactive ZKPs, the prover sends a single message (a proof) to the verifier. This is often denoted as $\pi$ (for proof). The verifier then checks the validity of this proof. The notation can include a statement like $V(x, \pi) = 1$ to denote that the proof $\pi$ is valid. As this thesis applies non-interactive proofs, we will not delve further into the discussion of interactive proofs, which require a back-and-forth communication between the prover and the verifier.

ZKPs can be characterised by three core properties [15]:

- **Completeness:** If a statement is true, then an honest prover can convince an honest verifier that they know the correct input.
- **Soundness:** If a statement is false, then no dishonest prover can unilaterally convince an honest verifier that they possess knowledge about the correct input.
- **Zero-knowledge:** If the statement is true, then the verifier learns nothing more from the prover other than the statement is true.

#### Beneftits of ZKPs

In summary, a key benefit of ZKPs is that the computation is verifiable. This allows for the confident assessment of computational correctness without revealing any underlying data. Additionally, a significant benefit of ZKPs lies in the efficiency of their verification process, making them highly practical for various applications where both privacy and computational efficiency are paramount.

#### Use Cases of ZKPs

The full potential of ZKPs has not been entirely explored, even four decades after their introduction. This section provides a brief overview of some current and potential use cases to illustrate where they are currently or could be applied today.

ZKPs are advantageous in proving statements on private data. For example, this can be applied to proving liquidity of bank accounts without revealing exact balances, giving access to websites without sharing login information or outsourcing of computations to scale blockchains [16]. In this case, the advantage is proving compliance with renewable energy consumption standards without revealing smart meter data.

Today, the primary use case of ZKPs is within the field of blockchain, where they facilitate private transactions and verifiable off-chain computations to reduce on-chain gas costs, which improves blockchain scalability. Furthermore, private data can stay confidentially off-chain and only the proof, proving the correctness of the computation, is written on-chain. This allows leveraging of private data in smart contracts without revealing the underlying data, which broadens the applicability of public blockchains in various business models.

### 2.2.2 Zk-SNARKS

Zk-SNARKs (Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge) [16] stand out within the ZKP family for their efficiency and non-interactivity. These proofs are more compact and quick to verify. The verification of zk-SNARKs relies on a trusted setup, which produces a common reference string (CRS). This CRS is divided into separate proving and verification keys in ZoKrates, which will be used in this thesis. The integrity of zk-SNARKs is dependent upon the secure generation of this CRS. Any compromise during its creation could potentially jeopardise the security of the entire system, making it possible to generate fake proofs. With this CRS, the prover can construct a proof that proves the correctness of a computation without exposing the actual data involved. The verifier has the verification key, which can then confirm the proof's legitimacy swiftly and with significantly less computational power than that required for proof generation.

### 2.2.3 Recursive SNARKS

Building upon these, recursive SNARKs [17] have emerged as an advanced development. Recursive SNARKs allow for the composition of multiple zk-SNARK proofs into a single proof, where "proofs are verifying other instances of themselves" [18]. This capability is beneficial in complex systems with multiple layers of data or processes that need to be verified. Recursive SNARKs enhance scalability and reduce computational load by streamlining the verification process to a lighter single proof, instead of multiple proofs or one complex proof. While the concept of recursion lacks an universal or general definition in the context of ZKPs, the existing literature agrees on one main property: the *incrementally verifiable computation* [19]. This refers to the construction and verification of a proof for each step of a computation process, where the verifier only verifies the final proof, that inductively proves the correctness of all other proofs.

### 2.2.4 Distinguishing Nesting from Recursion

While most of the existing literature employs the term "recursion" to describe a proof composition such as in this thesis, the term "nesting" will be employed here. The main reason for this is, because the core aspect of recursion in computer science is a form of *self-reference* [20], where for example a function calls itself. This inherently implies a more circular and interdependent structure, which can be seen in Figure 2.2, where the succeeding output is utilised in the previous or same component as an input again. Nesting, on the other hand, implies a more hierarchical structure, where one component is merely placed inside the other, without referencing itself, as can be seen in Figure 2.3. While both approaches create a layered architecture, nested layers do not depend on the execution of themselves.

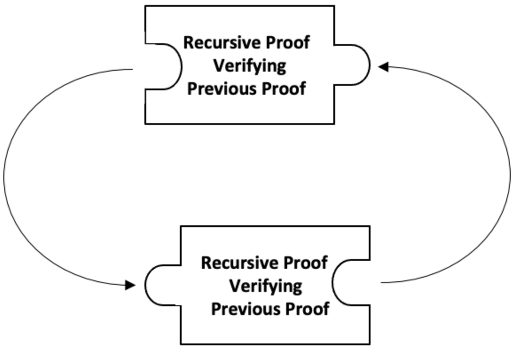

Figure 2.2: Conceptual illustration of two recursive proofs verifying each other through *recursion*

### 2.2.5 Benefits of Nesting

In summary, nesting describes the process, where one or more proofs, the inner proofs, are embedded within another, the outer proof. Importantly, nesting supports multiple inner provers, making it possible for a variety of provers to provide inner proofs simultaneously and thereby enhancing the efficiency of validating data from multiple sources. In doing so, collective validations become possible. Additionally, this nested structure enables a third party to verify just one single outer proof, which encompasses and confirms the correctness of all computations of its inner proofs. Nesting benefits both the prover and the verifier. The prover can reduce the computational load by splitting a

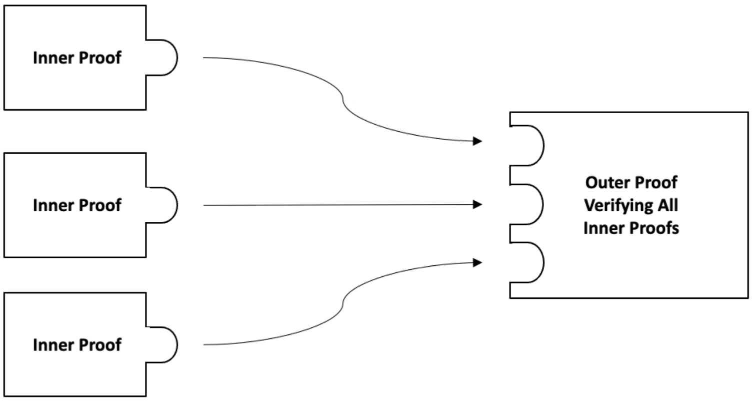

Figure 2.3: Conceptual illustration of one outer proof verifying multiple inner proofs through *nesting*

single and larger proof into smaller segments and computing each individually or even distributing the computation to multiple parties. This becomes beneficial in memory-restricted environments. Furthermore, different parts of a computation can be processed in parallel, speeding up the overall computation time. For the verifier, nesting allows the verification of a single aggregated proof, making it more cost-effective, especially for on-chain verification where verifying multiple proofs is more costly. Lastly, nesting enables modularity of proof construction, where different modules can be developed, tested and verified independently before being combined into a larger proof.

### 2.2.6 ZoKrates Toolkit

The open-source toolkit ZoKrates [21], a key component in this thesis, supports the generation and verification of zk-SNARKs, offering essential tools for deploying these proofs in practical scenarios. ZoKrates improves scaling and privacy for blockchains, primarily Ethereum, by facilitating verifiable off-chain computations with zk-SNARKs. In doing so private data stays confidential off-chain and only the proof for the correctness of the computation is written on-chain. ZoKrates simplifies the complex process of creating ZKPs by offering a high-level domain-specific language, a compiler as well as generators for proofs and verification smart contracts. There already exists a broad body of work [22–24] that leverages ZoKrates for various applications and protocols, demonstrating its practicality and effectiveness.

A key functionality of ZoKrates is the transformation of complex computations into arithmetic circuits, represented as a set of constraints. These constraints are used to generate a proof that verifies the computation's correctness without revealing underlying data. ZoKrates' efficiency lies in optimising these constraints, ensuring the proofs are succinct and computationally feasible. As a rule of thumb, less constraints lead to quicker computation time.

### Public and Private Inputs

Another important technical detail to note for this thesis' context is that computations in ZoKrates involve both private and public inputs. Private inputs remain hidden, ensuring the confidentiality of sensitive information. Public inputs, on the other hand, are visible and verifiable in the resulting proof.

### Negative Numbers

As ZoKrates operates over a prime field, which is a finite field with a prime number as modulus, negative numbers are represented as their positive equivalent in the finite field. For example, in a simple finite field with a size of 7, the number -1 is equivalent to 6 (since adding 1 to 6 yields 7, which is 0 modulo 7). Furthermore negative numbers are represented as a `field` type, because the prime fields are relatively large and smaller types like `u64` cannot accommodate them.

### On-Chain Proof Verification via Smart Contract

Another key feature of ZoKrates is the export of the proof in a solidity smart contract for Ethereum. This enables a decentralised and automatic proof verification on the blockchain and enables all interested parties to verify the proof. However, ZoKrates and Ethereum do no support the curve required for nesting, which prohibits on-chain proof verification. This may change with the Ethereum improvement proposal 3026 [25] in the future.

### Trusted Setup via Multi-Party Computation protocol (MPC)

The zk-SNARK schemes supported by ZoKrates require a trusted setup, to generate the proving and verification keys. This procedure generates data often referred to as *toxic waste* which can be used to create fake proofs which will be accepted by the verifier. Provers could exploit this process to generate false proofs, deceiving the verifier. Therefore the implementation of the trusted setup phase must be decentralised. This decentralisation can be achieved via a MPC, in which responsibility is distributed among all participants. This approach allows the verifier to eliminate their toxic waste and thereby ensuring the validity of all subsequent proofs. Unfortunately the current version of ZoKrates does not support MPC with the required elliptic curves for Nesting.

# 3 Related Work

Although studies specifically focusing on privacy-preserving methods for policy compliance in local renewable energy markets are scarce, there is a considerable body of research dedicated to the broader areas of privacy-preserving data aggregation and smart metering. This highlights a notable gap in research that specifically connects these areas to energy compliance. This section will focus on few selected studies to illustrate the spectrum of approaches in the field.

We will start by exploring more general privacy-preserving data aggregation protocols beyond the context of smart metering and then delve into protocols specifically tailored for smart metering, also in connection with ZKPs. Following this work regarding ZoKrates in the context of privacy-preserving data aggregation and its applicability to smart meter data will be discussed. Lastly, we will examine the a gap in existing literature concerning the topic of nesting. However, we can explore some practical work regarding the closely related concept of recursion in relation to ZKPs.

## 3.1 Methods in Privacy-Preserving Data Aggregation (PDA)

As previously introduced, the aim this thesis is to develop a system that allows effective and verifiable compliance checks at a collective level, while still protecting individual privacy. Therefore, it is essential to privately aggregate data. This objective parallels the goal of PDA from existing literature. PDA protocols have been introduced to aggregate data from multiple sources and calculate statistics on sensitive data without infringing upon privacy [26]. Even though not all the works discussed in this chapter are explicitly labelled as PDA, they are categorised as PDA, as they align with the definition from above. Bearing this in mind, this section discusses alternative approaches to PDA that do not necessarily rely on ZKPs, which highlights different potential technologies for achieving similar objectives.

### 3.1.1 Hypergraph-Based PDA

Dekker et al. [26] propose a PDA protocol, which employs a hypergraph-based detection algorithm, that probabilistically detects out-of-range user values without disclosing them to the aggregator. This protocol is suitable for aggregators working with resource-constrained users, as the hypergraph structure allows the aggregator to quickly pinpoint malicious users by examining the intersection of groups that deviate from the range. Furthermore, this protocol enables advanced algorithms like principal component analysis and decision tree classifications. However, the detection mechanism is inherently probabilistic, meaning it operates based on likelihood rather than absolute certainty. While the protocol offers strong privacy assurances, there exists a limit of colluding users beyond which privacy can be compromised.

#### Dekker's et al. [26] Remarks on ZKPs for PDA

They note that a shortcoming of much previous research is the assumption of *honest-but-curious* participants for PDA protocols, a model where parties are expected to follow the protocol but may attempt to learn additional information. Dekker et al. acknowledge the potential to overcome this limitation by transitioning to a *malicious model*, which can be achieved through the implementation of ZKPs. However, as traditional ZKP approaches rely on a trusted setup, such as zk-SNARKs, and the computationally intensive proof generation, render them less attractive for trustless or resource constrained scenarios.

### 3.1.2 Utilising Homomorphic Encryptions for Smart Meter Data

Alternatively, Mohamadali et al. [27] introduce a protocol for data aggregation in smart grids, that employs Paillier encryption. Their protocol has various features including batch verification, fault tolerance and support for multi-category data aggregation. These features enable more detailed insights into energy demand or load. While this protocol is efficient in terms of computational resources and secure under malicious models, it has a single point of failure, due to a centralised trusted authority that manages the whole system.

Similarly, Zhang et al. [28] utilise Paillier encryption for both temporal and spatial data aggregation in smart grids, to provide utility companies with the necessary information for billing and grid management. In this context, "temporal aggregation" refers to combining data points over time and "spatial aggregation" across different locations. While it is secure under malicious models as well, they acknowledge in their security analysis that their system depends on certain technical conditions to ensure privacy, such as requiring non-symmetric data matrices and multi-element user sets.

While there are several more studies that leverage homomorphic encryption for data aggregation in smart grids, this section briefly introduced those two studies, to illustrate this approach. However in summary, while these approaches allow for privacy-preserving computations on encrypted data, they may not fully address the more complex, multi-layered computation and verification processes that will be introduced in this thesis.

### 3.1.3 Utilising ZKPs for Energy Billing

ZKPs have been implemented to verify energy usage for billing purposes in smart grids, while ensuring that the actual consumption data stays confidential.

Rial et al. [29] propose a system where electricity customers can compute non-interactive ZKPs on personal devices, that verify whether their consumption falls in certain billing tariffs without disclosing any meter data. The system supports a variety of tariffs, including time-of-use tariffs. They demonstrate that a single computer could verify billing proofs from a three week period of 27 million households in 12 days, even in their slowest unoptimised version. However their performance evaluation considers only proof size along with proof and verification time, ignoring the complexities involved in setting up the proof system.

In a separate study, Jawurek et al. [30] introduce a plug-in privacy component, such as a router, which generates ZKPs based on Pedersen commitments. The proof confirms the consumers time-of-use tariff without disclosing any consumption data. Unlike the study of Rial et al. [29] this study is limited to a single tariff system.

More studies integrating ZKPs with smart metering will be discussed in later sections of this chapter where they are more contextually relevant.

### 3.1.4 Blockchains and Ledgers in PDA

Studies employing blockchains for PDA often combine these technologies with privacy-preserving methods, such as Bloom filters, homomorphic encryption, or ZKPs. This section will introduce some studies to provide an overview of the field.

Mouris and Tsoutsos' "Masquerade" protocol [31] offers a lightweight general-purpose protocol for verifiable data aggregation, optimised for computing aggregated statistics, such as histograms and averages. They are utilising Paillier encryption to encrypt private data, ZKPs to verify the correctness of the computations and the Fiat-Shamir heuristic to convert interactive ZKPs to non-interactive ones. Furthermore the protocol is adaptable enough to support smartphone clients in dynamic user environments. Despite its strengths, the reliance on a central entity managing the ledger introduces a potential single point of vulnerability and their protocol has not been peer-reviewed.

The study by Mahmoud et al. [32] contributes to the field PDA through its blockchain-based approach for Water Distribution Systems (WDS). It presents a method for aggregating smart meter data while providing user anonymity, applying blockchain to protect against data tampering and Bloom filters for identity verification. In doing so, the operational process becomes more transparent, as water quality and quantity or collected data, such as pipeline pressure, become verifiable. However the study acknowledges that their implementation faces immense challenges, because their blockchain's consensus algorithm relies on mining, which necessitates special mining nodes, that consume excessive energy and may not be suitable for most sensors, such as smart meters or hydraulic sensors.

Similarly, Guan et al. [33] encounter comparable challenges with their mining nodes to run blockchains for PDA in smart grids. Their approach involves dividing users into groups, each with its own blockchain that records data. Similarly again, they utilise bloom filters to authenticate users. This system faces challenges in balancing privacy, computational overhead and real-time data processing as well.

Homomorphic encryption is a popular choice, as multiple other studies also combine it with blockchain. For example, Fan et al. [34] utilise this approach along with leader election algorithms and Boneh-Lynn-Shacham short signatures to ensure data security in their blockchain system. However, these systems may as well be confronted with the limitations of employing homomorphic encryption in scenarios like discussed in this thesis, as discussed in Section 3.1.2.

Miyamae et al. [35] combine blockchain with ZKPs for renewable energy trading. They introduce a blockchain system with an UTXO token representing electricity production. The system employs zk-SNARKs to aggregate and compress smart meter data to 1/1000th of its original size, aiming to improve scalability and ensure privacy and data integrity on-chain. However, they express concerns about the potential for the total number of smart meter records to exceed the blockchain's capacity.

### 3.1.5 Utilising MPC

Previously in Section 2.2.6, MPC was introduced in the context of decentralising the trusted setup in ZoKrates, but it can be also utilised in PDA for smart meter data.

Kirschbaum's et al. [36] protocol introduces a MPC, where each smart meter participates in the computation of aggregated consumption data. Instead of exchanging individual consumption data, each smart meter uses a random number generation function to efficiently compute its contribution to the aggregated energy consumption data. This aggregated data, which integrates values from all meters without revealing individual meter readings, is then sent to a gateway. As a result, no participant in the network, including the gateway, can determine the individual consumption data of any specific smart meter, thereby ensuring privacy. While the system effectively aggregates data for analysis, they acknowledge it does not support the real-time control and management required in smart grid technologies.

Thoma et al. [37] have developed a similar system, which they also enhance with homomorphic encryption to enable additional features like real-time demand management. Besides using the MPC for PDA, this system enables each consumer send encrypted data to the utility for precise billing. It also includes a graphical user interface for a smart home control panel. This panel allows consumers to monitor their household's real-time electricity consumption, prices and billing, as well as track the electricity usage of individual devices.

### 3.1.6 Perturbation Techniques

There are alternative approaches for preserving privacy in smart metering, such as differential privacy [38] or noise addition [39]. While these method are effective for masking individual data within aggregated datasets, they fall short in performing detailed computations or condition verification on individual data. Therefore, these studies are not discussed in detail, but they remain nonetheless interesting contributions to the field of privacy preservation in smart metering.

## 3.2 Zokrates in Privacy-Preserving Systems

As this thesis utilises ZKPs created with ZoKrates, this chapter will continue exploring system leveraging ZoKrates for privacy preservation.

### 3.2.1 ZoKrates for PDA

Ismayilov and Ozturan's protocol [24] on the Ethereum blockchain uses ZoKrates proofs for PDA, focusing on secure summation of encrypted values. It utilises a pair of hypercube configurations for data aggregation to ensure privacy and security. Each participant's data is committed, encrypted and the integrity as well as the correctness of the aggregated data are verified using ZKPs with ZoKrates. The smart contract and user web interface support the functionality and provide a practical implementation of the protocol. While effective in maintaining individual data confidentiality and aggregate sum verification, it falls short in verifying more complex conditions within the aggregated data. Additionally they conclude that the communication overhead of their protocol needs to be reduced in future work.

### 3.2.2 ZoKrates in Other Scenarios

Adding a different perspective, Heiss et al. [22] introduce a novel approach to verifiable carbon accounting in supply chains using ZoKrates. Companies can generate ZKPs and deploy them on a blockchain to allow peer-to-peer verification of their carbon footprints without disclosing sensitive data. While the study is focused on carbon accounting, the use of ZKPs and zoKrates in this context offers insights applicable to the preservation privacy in energy compliance, demonstrating the potential of those in balancing verifiable environmental claims with privacy.

### 3.2.3 ZoKrates in Local Renewable Energy Markets

Eberhardt et al. [23] utilise ZoKrates for validating energy netting in local renewable energy markets, specifically to optimise the use of locally produced energy and reduce reliance on external power sources. Their method ensures individual household energy contributions remain confidential while verifying the community's overall energy balance.

By integrating ZKPs with commitment schemes, their approach achieves both privacy and computational correctness in the netting process. In doing so, the need for a netting algorithm to be executed directly on the blockchain is eliminated, as the algorithm is executed off-chain with ZKPs on a "Netting Entity". Instead, only a "Netting Verification Contract" is deployed on the blockchain that verifies the correctness of the end result. They propose a "Household Processing Unit" (HPU), that adds a more resourced layer to interact with the blockchain. The HPU registers signed meter readings in the netting smart contract through a blockchain transaction to make tampering evident and additionally provides a user interface for the residents. Notably, Eberhardt's et al. work aligns with this thesis' use of ZoKrates, by demonstrating its practicality for privacy preservation of energy data in local renewable energy markets.

## 3.3 Exploring Nesting of ZKPs

The concept of nesting of ZKPs has not been yet extensively explored in academic literature from an application or use case perspective. Even though the survey by Morais et al. [40] recognises the potential of enhancing privacy-preserving validations by aggregating multiple ZKPs, they stop from delving deeper into this concept. However the theoretical foundations that enable recursion and nesting, such as the underlying cryptographic primitives, have been deeply discussed in existing literature, as introduced in Section 2.2.3.

### 3.3.1 Recursion in Cryptocurrency Platforms

Beyond academic literature, the closely related recursion of ZKPs has been effectively implemented in cryptocurrency platforms such as Polygon [41]. Here they enhance transaction privacy and scalability, increasing throughput while simultaneously reducing latency. This is achieved by storing only one recursive proof on the blockchain as opposed to vast amounts of single proofs.

Another example is the Mina protocol [42], that utilises recursive SNARKs to enable network nodes to store only a succinct proof instead of the entire blockchain. This single proof allows for verification of the correctness of the whole blockchain. As the proof verification time is constant, it remains the same regardless of the blockchain's size.

Nonetheless, the broader applications of recursion and nesting go beyond the sector of digital assets and have yet to be fully explored.

## 3.4 Summary of Limitations and Gaps

Hypergraph-based PDA and homomorphic encryption methods face challenges in centralised vulnerabilities as well as limitations in more complex validations or condition checks. ZKPs in smart metering demonstrate privacy preservation but lack an exploration of more collective scenarios or modular verification. Blockchain implementations for PDA offer robustness but struggle with ledger management and computational efficiency. The use of ZoKrates for privacy-preserving systems has been demonstrated and highlights its capability for various applications. Lastly, the potential of nesting ZKPs in privacy-preserving systems remains largely unexplored, which highlights a gap in existing literature.

In essence, the limitations identified in related research, coupled with the underexplored area of nesting, show a promising research opportunity in applying nested ZKPs for privacy-preserving collective policy compliance in local renewable energy markets.

# 4 Research Question

The existing body of research offers valuable insights into privacy-preserving techniques across various fields, ranging from hypergraphs to blockchains. Despite the diverse applications of ZoKrates in areas such as carbon accounting or energy netting, a clear gap remains in applying these techniques to policy compliance in local renewable energy markets. Most methodologies encounter limitations in ensuring verifiable collective compliance or maintaining household privacy, particularly under malicious models. Additionally, the potential of nesting ZKPs, a promising approach for more complex and layered validations, remains largely unexplored in practical applications. Building upon these insights, this thesis aims to synthesise these areas of research, utilising ZoKrates-based nested ZKPs to privately aggregate data and verify compliance. In doing so this thesis aims to create a novel approach of privacy-preserving and collective renewable energy policy compliance, as can be seen in Figure 4.1.

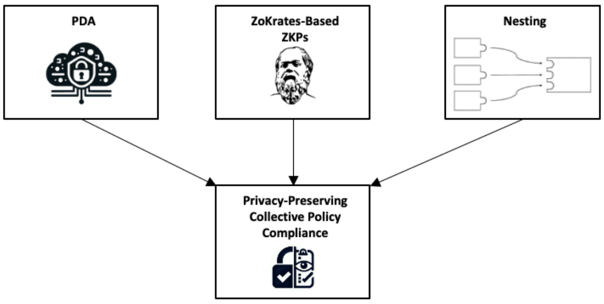

Figure 4.1: Mapping the synthetisation of diverse research fields in this thesis

Formulating the core objective of this thesis' proof of concept, energy data needs to be aggregated across households to verify collectively if sufficient renewable energy is consumed inline with policy thresholds. However, remembering Section 2.1.1, energy data without privacy protection can pose a serious problem for households. Nonetheless, transparency of the energy data is necessary to prevent deviant behaviour. Hence, the aim of this thesis is to develop a system that allows effective and verifiable compliance checks at a collective level, while still protecting individual privacy. In order to accomplish this, the nesting of ZKPs is applied to create verifiable and collective compliance checks in local renewable energy markets. The way nesting is applied here can be seen in Figure 4.2, where the local utility aggregates inner proofs from households, creating one comprehensive outer proof proving the neighbourhoods compliance with renewable energy thresholds, which an auditor can independently verify.

This leads us to the research question of this thesis: **How can we enable veri-**

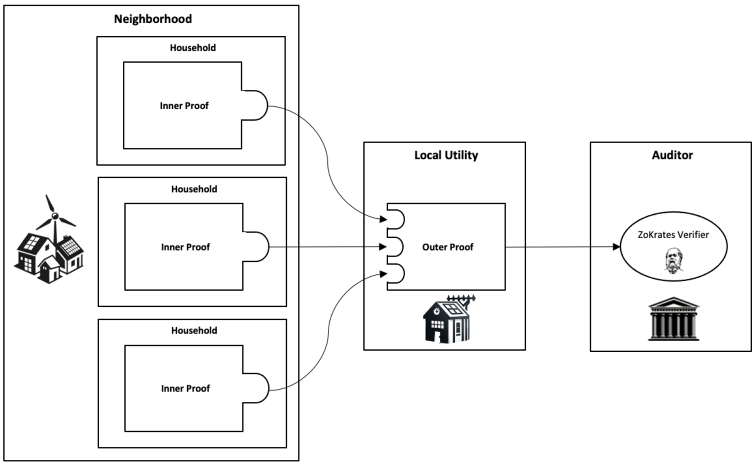

Figure 4.2: Schematic illustration of the application of nested ZKPs for privacy-preserving and verifiable collective policy compliance in local renewable energy markets

**fiable collective policy compliance in local renewable energy markets while preserving household privacy using nested ZKPs?**

Abstracting from the specifics of this particular use case, this thesis aims to present a fundamental result: it seeks to demonstrate how ZoKrates-based nested ZKPs can be effectively implemented to verify collective criteria, while simultaneously preserving individual privacy.

# 5 Evaluating the Socio-Economic Consequences

We will transition from the essential concepts and technologies discussed in the previous chapters to the technical part of the thesis, where the proof of concept is introduced. However, before diving into the more concrete system design chapter, it is important to understand and assess the broader impact and benefits that such a system potentially introduces. Bearing this in mind, this chapter provides a more holistic and socio-economic exploration of the system's potential impact.

## 5.1 Assessing the System's Multifaceted Benefits

This section examines how the system not only addresses the immediate objectives but can also contribute positively to a wider context. An analysis is conducted from a variety of perspectives to provide an understanding of its potential value and significance.

### 5.1.1 Reducing State Surveillance and Intrusion

As noted in Chapter 1, it can be seen that the global trend towards state monitoring and data collection has increased in recent years [3]. Furthermore, the trend towards environmental regulation and compliance can be observed to be increasing as well [4]. Traditional compliance methods often require extensive data collection and monitoring by state agencies, which can intrude into private lives. Those two considerations together highlight the need to develop systems that protect individual privacy while still fulfilling necessary societal functions.

In this context, a system that enables privacy-preserving compliance verification can reduce state intrusion, as the state can complete its task of enforcing compliance or regulation, such as renewable energy standards, without overarching surveillance and intrusion into private lives.

Consequently, the need for physical inspections and compliance verification is also reduced, as the automation and digitisation of this whole process allows to minimise the frequency of physical inspections. This is not only more efficient and less bureaucratic but also aligns with the societal need towards less intrusive governance.

### 5.1.2 Setting a Precedent for Privacy-Preserving Compliance

In essence, such an system can exhibit that compliance verification does not have to compromise on privacy. This system can set a precedent for future regulatory policies, by demonstrating how regulations can be designed to achieve their objectives without excessive state surveillance. With the implementation of privacy-preserving compliance and regulation, the public perception of regulation may shift towards a more positive narrative and lead to greater public acceptance and cooperation in regulatory matters.

### 5.1.3 Empowering Communities and Encouraging Collective Responsibility

The concept of *collective energy practices* introduced by Verkade et al. [43] provides a profound framework for understanding how community-driven initiatives can empower communities and individuals. These practices are geared towards shared and not solely individual goals. Within this thesis' context, it encourages community autonomy and responsibility by giving the control over compliance and energy usage to communities, shifting the focus from a state-centric monitoring to community-driven initiatives. This empowers communities to work collectively to achieve compliance to the renewable energy targets, which can encourage collaboration and foster a sense of togetherness. In contrast, focusing on individual compliance may increase social inequities, as it would require households with fewer financial resources to meet the same renewable energy production targets as wealthier households. Furthermore, Moret et al. [44] discuss how energy collectives can democratise energy systems, optimise community energy resources and enhance fairness within communities. While they focus on community-level management and optimisation of energy resources, delving deeper into this idea reveals within the context of this thesis, that a collective approach enables wealthier households to assist less affluent ones. By participating more substantially in renewable energy investments, wealthier community members can help the community to achieve the renewable energy consumption targets. In doing so, both equity and sustainability are improved, by sharing the responsibilities across the community.

### 5.1.4 Encouraging Renewable Energy Transition Through Compliance Verifiability

The system encourages the transition towards renewable energy, by enabling an efficient monitoring of compliance without infringing upon privacy. Moreover, by enabling compliance verification in local renewable energy markets, the mere existence of a compliance process and an authority leads to increased compliant behaviour, even without direct monetary benefits or explicit threats of sanctions [13]. Thus, households are incentivised to meet their renewable energy targets and achieve compliance. This transition will not happen overnight, but being pressured for non-compliance by auditors and the general public can promote investments into solar panels or other renewable sources. As seen in Chapter 1, a majority of carbon emission stems from household energy consumption and transitioning those to renewable energy is a great step to target climate change.

### 5.1.5 Providing Data of Regional Dynamics in Energy Compliance

As data is collected about the compliance of communities, it can be utilised to analyse the influence of policies and regulations in different geographical regions. This data may serve as a valuable resource for policymakers and allows them to identify patterns and conditions that distinguish successful regions from non-successful ones. Based on this data, policymakers can initiate various initiatives, such as infrastructure projects and other measures, aimed at helping non-compliant regions transition towards renewable energy.

## 5.2 Assessing the System's Multifaceted Costs

While acknowledging the potential benefits associated with this thesis' privacy-preserving system, it is equally important to recognise the potential costs. In the following discussion, we will delve into the various costs associated with this system, excluding straightforward hardware expenses such as those related to smart meters.

### 5.2.1 Economic Value and Welfare Consequences of Privacy

Rust et al. [45] provide an economic model, that suggests that maintaining privacy is becoming increasingly expensive over time, but can always be purchased on "emerging markets for privacy". However, even with privacy-preserving technologies, systemic pressures might reduce privacy levels over time. In the context of this thesis it means that the costs associated with maintaining such a system will always be more expensive than not preserving privacy in compliance verification. This can potentially divide communities, as not all members value privacy economically equally, which can erode the whole community cohesion, if only a minority of the community values privacy enough to "purchase" it.

Adding a different perspective, Acquisti et al. [46] argue that privacy protection can lead to societal welfare loss, as potentially insightful data cannot be analysed to benefit society as a whole. While they acknowledge that "balancing privacy and disclosure" is context-dependent, a more in-depth research is needed to understand how the consequences of privacy impact societal welfare in the context of local renewable energy markets. Nonetheless, their consideration offers valuable insights for the broader discussion within this chapter.

### 5.2.2 Technical Issues and Ensuring Reliability

Ensuring that every component of the system is working reliably and connected poses challenges. Typical residents may lack the knowledge or experience to guarantee that the system's components are functioning correctly. In scenarios where some households fail to submit their proofs, the aggregated outer proof may not be sufficiently expressive to demonstrate neighbourhood compliance. For instance, if residents accidentally turn off the components located in their household, there should be a mechanism in place to resolve this problem. A system administrator might need to resolve such problems, however it must be acknowledged that such a person would require the necessary training. Additionally, ongoing maintenance and support for the system and smart meters can present challenges, particularly in ensuring that all technology and staff remain up-to-date and continue to operate effectively over time.

#### Synchronisation of Data Collection

All participating households within a community must align their data collection processes to the same timeframes. This is essential to ensure that the data reflects a consistent and accurate picture of energy consumption for the community. If households submit their data from different periods, the internal community renewable energy balance is not accurately measured. If a household's component like the smart meter is offline for a couple of days, the household needs to synchronise again with the rest of the group, which has the potential to be a complex task.

### 5.2.3 Establishing User Compliance and Behavioural Challenges

There is a risk that residents might not fully engage with or potentially misunderstand the purpose and function of smart meters and the compliance system. If residents are unaware of the benefits or are skeptical of the technology, it can diminish the system's effectiveness if a majority of households do not want to join the compliance system. This skepticism can be deepened by privacy concerns, as highlighted in the study by Mani et al. [47], where the *Big Brother effect*, which describes the fear of pervasive surveillance, significantly influences user resistance to smart technologies. The study suggests that such fears of constant monitoring and data misuse can lead residents to disengage from or actively resist using smart meters and corresponding systems. This underscores the necessity for clear communication and education of residents about the system's privacy measures and its benefits to mitigate concerns and enhance the system's adoption.

### 5.2.4 Targeting Educational and Knowledge Barriers

In "The Economics of Privacy" [46], the authors argue that systems designed to preserve privacy require additional user engagement and expertise, potentially undermining the system's effectiveness, especially among some of the most susceptible groups within society. It is therefore essential to sufficiently educate and train the residents, allowing them to understand the system's benefits and utilisation. Additionally, staff or specially commissioned residents need to be trained to support the households and for example resolve issues with smart meters or other software components. Understanding ZKPs might also be complicated even to technical affine people, which requires the creation of qualitative educational material and courses.

# 6 System Design

With the research question in mind and the more abstract exploration of the socio-economic consequences this system potentially introduces, we can now delve into technical details of the proof of concept. This chapter will continue with outlining the system's design, including the roles of involved stakeholders and their interactions. It will further present both the functional and technical requirements for the system's operation, followed by the architectural composition of its components. The chapter concludes by evaluating potential risks, analysing their impacts on the system and outlining strategies for their mitigation.

## 6.1 System's Goal

The system's goal is to allow for external parties to verify if collectives meet certain conditions without revealing their sensitive data. This thesis introduces a proof of concept for such a system in a specific use case: verifying the collective compliance of neighbourhoods to renewable energy policy standards in a privacy-preserving manner. In order to allow for the preservation of privacy and disallow its misuse, the system needs to have mechanisms to prevent deviant behaviour.

## 6.2 Operationalising the Goal

This goal is realised through nesting of ZKPs, allowing for an auditor to verify one proof that proves whether a neighbourhood uses enough renewable energy, in line with a predefined threshold, all while safeguarding each household's privacy. Internally, this single outer proof aggregates and verifies an inner ZKP from each households, which contains the relevant energy data and proves its authenticity and integrity.

## 6.3 Stakeholder Functions and Interactions

Four key actors are involved in this use case: the *households* consume sufficient renewable energy with the goal of meeting a renewable energy consumption threshold. In the proof system, they act as provers, providing the inner proofs that prove their energy readings are not tampered with. The *local utility*, besides being the central energy supplier, aggregates all inner proofs from the neighbourhood and utilises them as inputs for the outer proof. In doing so, it acts as well as a prover in the proof system. The *auditor*, which can be a governmental body, acts as a verifier in the proof system that verifies this aggregated outer proof to ensure the neighbourhood aligns with the established renewable energy consumption threshold. The *smart meter operator*, while not participating actively in this system, is supplying and maintaining the smart metering technology. Lastly, a *system administrator* supports the households to ensure every component is working properly and resolves potential failures. This role can be taken on by a employee of the local utility or a resident of the community.

### 6.3.1 Trust Assumption

However, some degree of trust needs to be placed in the local utility to act honestly and not reuse a valid inner proof for the purpose of creating a deceptive outer proof, when the neighbourhood does not meet their renewable energy consumption threshold.

## 6.4 Functional Requirements

Having outlined the system's goal and the key stakeholders involved, we can examine the functional requirements that constitute the system's core operational capabilities and guide the development and implementation.

### 6.4.1 Privacy Preservation

The system's core objective is to preserve privacy, yet defining the corresponding requirement is complex. Leveraging the previously introduced principle of least privilege in local energy markets serves as a beneficial framework to guide this requirement's definition, as referenced in Section 2.1.1. Applied to this system, it means that the auditor should have access only to the data necessary to complete the task of verifying the community's compliance to renewable energy compliance targets. This involves that the auditor is aware only of the predefined threshold of their respective compliance policy and the neighbourhood's compliance status.

#### Optimising Data Transmission Frequency for Privacy Preservation

However, the local utility, acting as the aggregator, has the capability to access the household's energy data to some degree during the outer proof's construction. It might already have information about the household's energy consumption, given that it supplies energy to the households. Nevertheless, the risk of privacy infringement during proof construction should be minimised. The expressiveness through the energy's data frequency of transmission should be minimised, as highlighted in Section 2.1.1. Therefore the energy data utilised in the outer proof should not be sent too frequently, to ensure that the local utility cannot extensively analyse or utilise the data for purposes other than intended.

For that reason, I propose that the system adopts a semi-annual schedule for proof generation. From an analytical perspective, semi-annual proofs provide enough data points to capture variations in energy consumption and production, even in areas with significant seasonal fluctuations. Renewable energy sources like solar and wind power are heavily dependent on weather and seasonal conditions. For policymakers, this data provides a sufficient comprehensive overview of how well renewable energy targets are being met in the context of varying seasonal conditions. Thereby, the data is detailed enough to account for seasonal-related fluctuations while minimising the potential of analysing the energy data in depth.

### 6.4.2 Exposing Tampering and Enhancing Data Integrity

Given that households might try to avoid meeting the renewable energy thresholds, the system should be designed to make tampering evident. Reflecting on insights from Section 3.1.1, the system should operate reliably within a malicious model, where participants may behave deviant. On the hardware side, it is essential that the smart meter operator deploys smart meters that are designed to prevent unauthorised alterations to their readings or other components. Regarding the software side, the local utility needs to be sure, that the energy data aggregated from households has not been altered after leaving the smart meter. Outsourcing this process to other third parties, is not optimal. While the smart meter operator is tasked with achieving the measurement's integrity, assigning them the additional responsibility of managing household data introduces potential privacy risks. Thus, data integrity needs to be achieved at the households premises, which requires the system to be designed in a way that makes any tampering evident, while still preserving the privacy of the household.

### 6.4.3 Expressiveness of the Outer Proof

It needs to be ensured, that the outer proof is expressive enough to enable policy and decision makers to make informed decisions or assessments. This means that the proof should display whether renewable energy consumption and production aligns with the thresholds set by policies. The goal is to provide interpretable and actionable information to the auditor, enabling the option of providing rewards and punishments in the compliance system, as explained in Section 2.1.2.

### 6.4.4 Allowing Flexibility of Energy Thresholds

Given the variability and unpredictability of renewable energy sources like solar or wind and their uneven availability across different locations, it is essential to implement flexible renewable energy thresholds in the system. Having flexible thresholds allows the system to accommodate this variability and ensures that households are not unfairly penalised for fluctuations in renewable energy production that are beyond their control. Moreover there is no one-size-fits-all approach to thresholds for all locations due to different energy profiles in those. Allowing flexibility in the system ensures fairness and adaptability, which is crucial for gaining trust and support of households. Also, compliance standards may change over time, which may require changes to the thresholds.

### 6.4.5 Community Adaptability and System Flexibility

Due to the inflexibility of the proving system's underlying circuit, the outer proof cannot be generated with a missing inner proof. A solution has to be implemented to allow the system to function further in case an inner prover ceases to exist. Since neighbourhoods or communities do not rapidly change in size, households joining or leaving is not a primary concern. Nonetheless, the system should allow some households to join or leave the compliance community, without needing a completely new setup each time such changes occur. This flexibility ensures the system remains efficient and adaptable to minor community shifts.

## 6.5 Hardware Components and Technical Requirements

Following the definition of the functional requirements, we can identify the technical specifications necessary to actualise the envisioned functionality.

### 6.5.1 Smart Meters

Each household should have smart meters that are capable of measuring both the energy consumption and renewable energy production from own sources. The smart meters are trusted to function reliably and cannot be manipulated by the household. This might be achieved by incorporating a secure element into the meter, to resist physical and logical attacks and thereby prevent tampering. Furthermore, the smart meter should be able to cryptographically sign the energy readings to make any manipulation of the readings after transmission evident.

### 6.5.2 Household Processing Unit

Given that smart meters may not have the capabilities to create ZKPs with tools like ZoKrates, a more resourced layer is needed. Similar to the HPU introduced by Eberhardt et al. 3.3, an additional component is introduced that generates the inner proof for each household using ZoKrates. This processing unit, which can be a consumer-grade computer, is able to receive the signed smart meter readings and include those as inputs in the inner proof. The processing unit should also be able to compile and compute ZoKrates proofs. Lastly, it needs to transmit the inner proof to the local utility.

### 6.5.3 Local Utility's Processing Unit

Similarly, the local utility should be able to aggregate all inner proofs and construct an outer proof using those. The local utility needs to have sufficient computational resources to compute the outer proof using ZoKrates. Lastly the local utility should be able to export the proof onto a blockchain or transmit it directly to the auditor.

### 6.5.4 Auditor

Finally, the Auditor should be equipped to verify the outer proof by interacting with an on-chain smart contract or using ZoKrates.

## 6.6 Architecture and Workflow

With the requirements defined, the following section introduces the essential architectural components of the system and outlines their respective functionalities and interactions. In the context of this bachelor's thesis, which aims to present a proof of concept, certain aspects such as the specific methodologies for transmitting proofs between system components are not elaborated in detail. The primary focus is on demonstrating the feasibility and core principles of this system. Bearing this in mind, the system operates in four distinct stages:

### 6.6.1 Smart Metering Stage

The foundation of the neighbourhood's energy compliance is energy data. Therefore each household's smart meters measure the consumption and production of energy for a given interval. These readings are then signed, to make tampering evident after they leave the smart meter. The signature has to be on the same elliptic curve in which the inner proof is compiled to allow for signature verification inside the proof. Those signed readings are then transmitted to the processing unit to be utilised in the proof's computation.

### 6.6.2 Inner Proof Generation

The essential element for achieving privacy in balance with transparency is the inner proof. As discussed in the previous Section 6.4.2, the system aims to preserve privacy while simultaneously making tampering evident to avoid the misuse of privacy. In this context, the inner *proof of integrity*, denoted as $\pi_i(C_i, P_i)$, verifies the smart meter's signature on the consumption $C_i$ and production $P_i$ readings to ensure their integrity. Any alteration of energy data would result in a faulty proof, making tampering evident.

This step enables the energy data to be provided as a private input into the ZoKrates program, mitigating concerns about their integrity. Nonetheless, the outer proof requires access to some form of energy data to assess the community's renewable energy consumption. This is why a net result $N_i$ is provided alongside as a public input to allow the outer proof to extract the household internal energy net result, which is derived from the household's total consumption $\sum C_i$ minus the total production $\sum P_i$. This aggregated metric, reflecting the net energy usage for a period, ensures that detailed energy patterns remain confidential, thereby addressing the privacy concerns outlined in Section 6.4.1. After the inner proof is computed within the processing unit, it can be submitted to the local utility to be utilised in the outer proof's construction.

### 6.6.3 Outer Proof Generation

Upon aggregating all inner proofs, the local utility can compute the outer *proof of compliance* $\Pi$. Each inner proof $\pi_i$ and corresponding verification key is provided as a private input in the outer proof, ensuring the household's proof is not discernible from the outer proof, especially the inner proof's public net result $N_i$. Inside the ZoKrates program, each inner proof is verified and its net result extracted. In doing so, the outer proof can calculate the community net result $\sum_{i=1}^{n} N_i$ and check, if the threshold $T$ is not exceeded. The outer proof's $\Pi\,(T, \pi_1(C_1, P_1), \pi_2(C_2, P_2), \ldots, \pi_i(C_i, P_i))$ renewable energy assessment is mathematically expressed as:

$$\sum_{i=1}^{n} N_i \leq T$$

Moreover, by providing the threshold as a public input, the requirement from Section 6.4.4 is met, as the threshold can be changed with each proof generation and the auditor can verify the threshold. The system's proof composition is depicted in Figure 6.1.

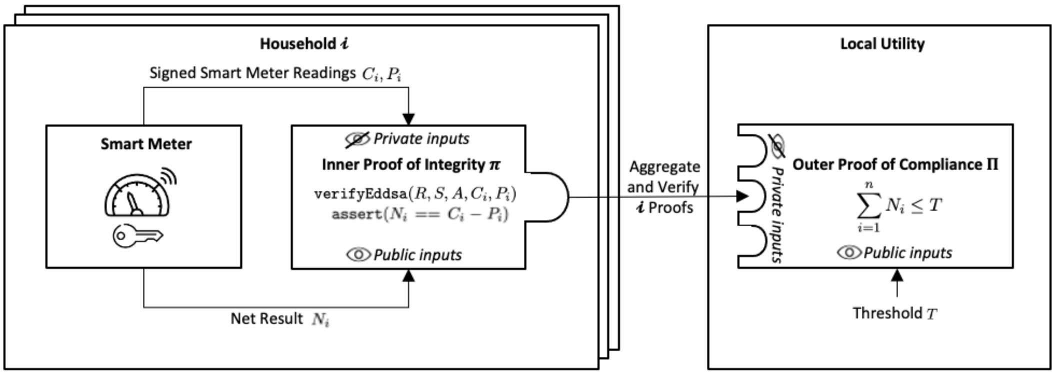
Household $i$
Signed Smart Meter Readings $C_i, P_i$
Smart Meter
Private inputs
Inner Proof of Integrity $\pi$
verifyEddsa$(R, S, A, C_i, P_i)$
assert$(N_i == C_i - P_i)$
Public inputs
Net Result $N_i$
Aggregate and Verify $i$ Proofs
Local Utility
Private inputs
Outer Proof of Compliance $\Pi$
$\sum_{i=1}^{n} N_i \leq T$
Public inputs
Threshold $T$

Figure 6.1: Conceptual representation of the system's proof composition

#### Allowing Shifts in Community Size

If a household decides to leave the compliance community, it can simply do so by ceasing to submit its inner proof. Conversely, if a new households wants to join, it must perform a new trusted setup with the local utility in order to participate. This requirement is specific to the current system implementation, which employs a trusted setup.

Incorporating additional inner placeholder proofs than the actual number of households in the outer proof enables the system to stay flexible under varying numbers of households. This feature allows the system to replace these placeholder proofs with an actual inner proof in the future, if for example new houses get build. This enables a more dynamic joining and leaving without requiring a new setup for the outer proof, but increases on the other hand the outer proof's constraints. Without this feature, the system would be limited to replacing existing households with new ones, thereby keeping the total number of participants constant.

### 6.6.4 Outer Proof Verification

Normally, ZoKrates proofs can be exported in a smart contract to the Ethereum blockchain. However, the elliptic curve required for nesting is currently not supported by any blockchain, see Section 2.2.6. Therefore, the auditor verifies the outer proof of compliance $\Pi$ using the `zokrates verify` command in an off-chain ZoKrates environment.

## 6.7 Risk Analysis and Contingency Planning

In addition to this chapter, this section determines potential vulnerabilities and risks of the system's architecture that might appear. Furthermore, the impact of potential failures is assessed and potential mitigation strategies are discussed.

### 6.7.1 Smart Meter Failure

The smart meters are essential for the proof system to capture the household's energy data. In case of a smart meter failure, the system lacks critical energy data, potentially skewing the neighbourhood's energy compliance. There are several reasons why a smart meter can fail, from simple hardware defects to complete energy outages. As the smart meter operates daily, unlike other components in the system, prompt resolution of the problem becomes crucial. There exist some protocols [48] that can detect smart meter outages and mitigate their failures effectively, which can potentially be applied in this context. Moreover, short-term outages of a couple of days might be tolerable in the long-term threshold assessment, as a margin of error can be adapted in the threshold to account for potential small scale failures.

### 6.7.2 Inner Prover Failure

In contrast to intermittent gaps of energy data caused by smart meter failures, not submitting an inner proof would lead to data absence from a whole household over the compliance schedule. However, as opposed to the smart meters, generating the inner proofs in this scenario is not time sensitive and does not happen frequently. Assuming that the smart meter functioned correctly, but merely the inner proof was not submitted, because for example the processing unit was unintentionally shut down, the system administrator has to contact the affected household and resolve the problem. Upon resolving the problem, a delay of the inner proof's submission is tolerable due to the neighbourhood's renewable energy compliance's non-urgent nature. Even a delay of weeks might be tolerable for the outer proof's generation. However, if the inner proof is persistently unavailable, it necessitates revising the thresholds, which will be discussed later.

### 6.7.3 Outer Prover Failure

The reliance on a single outer prover introduces a point of failure within the system. If the outer prover fails, the whole system becomes obsolete. Consequently, it is necessary, that the local utility prioritises the robustness of the outer proof's generation. This includes ensuring critical components, such as the proving key, are securely backed up to prevent system paralysis in scenarios of for example hardware compromise or damage.

### 6.7.4 Additional Threshold Requirements

Within the context of contingency planning, additional requirements for the renewable energy threshold can be identified. Firstly, the threshold should include a margin of error, which allows the system to accommodate minor smart meter outages without effecting the overall assessment. Secondly, the threshold needs to be flexible for a varying number of participating households, in order to account for flexibility of community dynamics or failures of inner provers. This can be achieved by designing the threshold based on an average renewable energy consumption value per household, which adjusts proportionally to the total number of households.

# 7 Implementation

Following the conceptualisation of this thesis' proof of concept, this chapter introduces the implementation of the nested proof system for privacy-preserving verification of renewable energy compliance. It features the components of the system, a mock smart meter and the generation of the inner and outer proof, as can be seen in Figure 7.1. Due to the lack of blockchain support for the elliptic curve required for nesting, a mock verifier is included. Furthermore, other challenges are addressed, like handling negative energy balances and adapting to dynamic household participation.

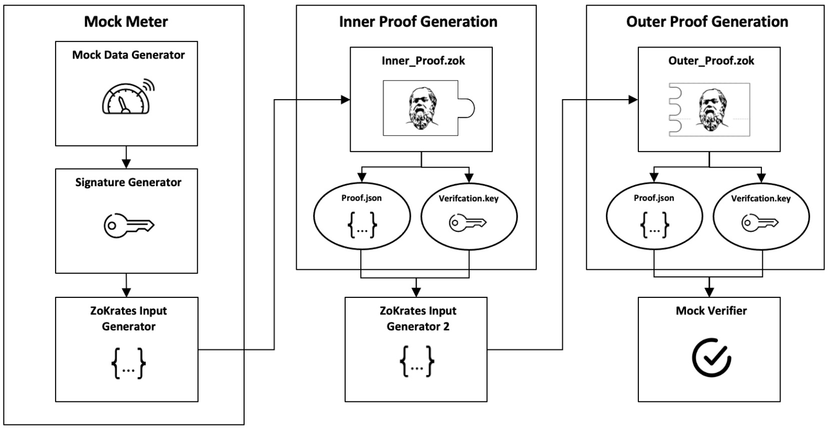

Figure 7.1: Schematic illustration of the implementation's components and their interactions

## 7.1 Mock Smart Meter

Inspired by Eberthardt's et al. [49] "mock sensor for decentralized energy trading", a mock smart meter has been implemented in Python that provides energy data and generates the required input file that is utilised in the inner proof's generation. The implementation produces mocked energy data based on Gaussian regression to simulate energy data measured in kWh. Furthermore this energy data is then formatted and packed into the correct input format for ZoKrates.

### 7.1.1 Signature Generator

A peculiarity is, that a specific signature is required that is compatible with the elliptic curve used in the inner proof's computation. Therefore a special implementation of the

ZoKrates PyCrypto package [50] is utilised to sign the energy readings on the decaf377 curve.

## 7.2 Inner Proof of Integrity

After the mock meter generates an `inputs.json` file, the inner proof of integrity is computed using this file. Listing 7.1 shows pseudo code for the inner proof, in which the smart meter's signature is verified and the correctness of the publicly provided net result ensured.

Listing 7.1: Pseudo code of the inner proof of integrity

```
def main(private R, private S, private A, private consumption, private
    production, public netResult){

    assert(verifyEddsa(R, S, A, consumption, production);

    calculatedNetResult = sum(consumption) - sum(production);

    assert(calculatedNetResult == netResult);

}
```

In the implementation shown here, merely one signature is applied for the purpose of simplicity and demonstrability. However, it can be changed to an arbitrary number of signatures and energy data. The evaluation chapter will examine the performance of different amounts of signatures. The full implementation in ZoKrates can be seen in the Annex.

### 7.2.1 Trusted Setup Phase

Initially, a trusted setup is necessary for generating the proving and verification keys, among other components, to establish the proof system. While the focus of this thesis does not lie on the setup phase of ZKPs, it is worth explaining the different setup phases that are needed in this implementation. Theoretically, all households, the local utility and the auditor could compute a MPC setup to decentralise the trusted setup phase. Unfortunately, as of now, ZoKrates does not support the necessary elliptic curves for nesting in its MPC protocol. As a result, this implementation requires for `n` households `n+1` separate trusted setups, as an additional setup between the local utility and the auditor is needed.

### 7.2.2 Smart Meter's Signature Verification

ZoKrates' standard library does not support the for nesting required BLS12_377 (decaf377) curve for signature verification. Therefore, Mehrpoya's implementation of the decaf377 curve in ZoKrates [50] is utilised and the corresponding `verifyEddsa` function imported that allows the verification of the smart meter's signature inside the inner proof.

The verifyEddsa function parameters require two u32[8] arrays, which the mock meter provides and initialises with eight daily consumption and production energy readings. It is also possible to initialise less indices, such as seven, but in this implementation the full data capacity allowed by the function will be utilised. Consequently, in a scenario with a semi-annual schedule, as proposed in Section 6.4.1, 23 signatures are required within the inner proof.

### 7.2.3 Internal Netting to Improve Efficiency and Privacy

The variable netResult is the sum of all production readings subtracted from the sum of all consumption readings. In doing so, the inner proof has only one public parameter of type field, instead of separated public parameters of type u32[8] for each signature or energy data array. In doing so, the input array size of the resulting proof is minimised. In this implementation the outer proof's PROOF_INPUTS variable is initialised with one (for one field variable), while the alternative approach would require a significantly higher initialisation. This enhances the outer proof's performance notably. Moreover, the influence of the inner proof's input array size on the outer proof's performance will be shown in more detail in the next chapter.

### 7.2.4 No Hashing to Enhance Efficiency

As can be seen, the energy data is not hashed. This decision was made because the outer proof requires access to the energy data. By refraining from hashing the data, the inner proof can compute an internal net result and compare it against the publicly provided one, thereby making tampering evident. If the hash is provided, the inner proof would have to hash the energy data to compare it with the provided hash and thereby increasing the complexity and constraints of the proof. Furthermore, hashing the energy data is not required for the purpose of suitable formatting for the verifyEddsa parameters, as u32[8] arrays can be directly provided.

### 7.2.5 Providing the Public Key Privately

In the documented implementation of the imported verifyEddsa function, the public key is provided as a public input. This allows for verification of the public key's correctness. On the other hand this hinders the scalability of the outer proof, as it increases the array size of the inner proof's inputs for two indices. This implementation therefore assumes that the signatures and public keys originate from the smart meter and are not altered, in order to enhance the outer proof's scalability. Alternatively, the smart meter's public key can be hardcoded inside the inner proof to ensure its correctness.

### 7.2.6 Negative netResult in the Inner Proof

In case that a household produces more renewable energy than it consumes, the netResult becomes negative. Since ZoKrates cannot handle negative numbers directly, the smart meter needs to transform the negative netResult before providing it as input by adding it to the prime value of the underlying prime field of the utilised curve, as has been explained in Section 2.2.6.

## 7.3 Outer Proof of Compliance

The outer proof of compliance is shown in Listing 7.2 as pseudo code. It takes in the inner proofs from the households and verifies their correctness as well as the aggregated compliance with the threshold. The final implementation in ZoKrates can be seen in the Annex as well.

Listing 7.2: Pseudo code of the outer proof of compliance

```
def main(private proof, private verificationkey, public threshold) {

    for num_proofs {
        assert(verify(proof, verificationkey));
        totalNetResult += proof.netResult;
    }
    assert(totalNetResult <= threshold);
}
```

### 7.3.1 Inner Proof Verification

The outer proof needs to have access to the verification key and the `proof.json` of the inner proofs. The *ZoKratesInputGenerator2* collects all `proof.json` files and `verification.key` files from all households and streamlines them into a suitable json format that can be utilised to compute the outer proof.

### 7.3.2 Negative `netResult` in the Outer Proof

The challenge is that the outer proof utilises a different elliptic curve than the inner proof. To calculate a total `netResult` of the community, the inner proof's negative `netResult` needs to be converted to a negative number on the outer proof's elliptic curve. Therefore an additional function was implemented to convert a negative number from the Bls3_77 curve to a negative number on the bw6_761 curve. This function does not work for positive numbers due to the intricacies of the underlying prime fields, as positive numbers are the same on both curves until reaching the maximum of the smaller prime field of Bls3_77. Thus, a logical limit, which the energy data will not exceed, is implemented in the function. A value exceeding this limit is considered negative. The function can be seen in the Annex.

### 7.3.3 Joining and Leaving of Households

Since variables have to be fixed-length at compile time in ZoKrates and cannot be dynamic based on the provided inputs, a `NUM_PROOF` variable had to be defined before compiling to make the loop size fixed.

A limitation of the fixed loop size is the system's reduced adaptability to households joining and leaving the compliance community. Each time more household join than leave requires the outer proof to be recompiled, reducing the usability of the system, since a completely new setup needs to be performed. A potential solution to a dynamic size of participating households is expanding the `NUM_PROOF` variable to a higher number and using placeholder proofs with a `netResult` of zero for non-existent proofs. This enables future households to join without necessitating a new setup of the system. However, this comes with efficiency trade-offs and an increased number of constraints, as the placeholder proofs need to be verified every time as well.

### 7.3.4 Mock Verifier

Since the nested proof compiled on the bw6_761 curve is not verifiable on Ethereum today, a mock verifier in Python is also implemented, that verifies the proof inside a ZoKrates environment.

# 8 Evaluation

In this chapter the implementation of the nested proof system is evaluated. The evaluation included measuring the system's scalability and data handling capacity. This includes measuring the maximum number of inner proofs it can support within a given compiling time and assessing the volume of energy data it can process and verify.

The evaluation was conducted on a consumer laptop with a 1.6 GHz Dual-Core Intel Core i5 chip. Furthermore, Python 3 was utilised to execute the evaluation tests and the results of those can be seen in the Annex.

## 8.1 Inner Proof's Scalability

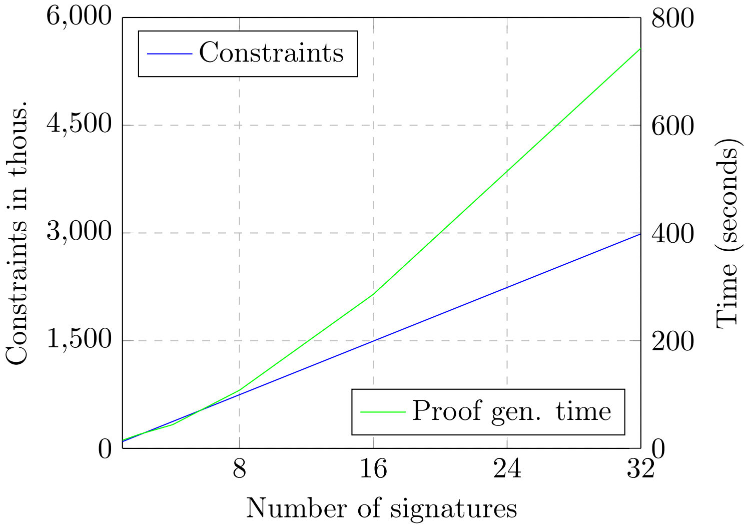

Figure 8.1: Comparison of constraints and proof generation time for various numbers of signatures

The inner proof's scalability was assessed by evaluating the impact of the amount of signatures and energy data. For each signature, the amount of energy data increases for eight consumption and eight production readings.

Figure 8.1 shows the effect of different quantities of signatures on the inner proof's constraints and computational performance. Increasing the amount of signatures results in a linear increase of constraints. The computation time increases linear as well, however, at eight signatures it accelerates, showing exponential growth. This exponential growth in the inner proof's computation time highlights ZoKrates scalability issues. While computational performance may improve with better hardware, the clear exponential trend indicates persistent scalability limitations. Furthermore, this limits the possibility of outsourcing larger computations of the outer proof into the inner proofs. Adding more layers of proofs to split the computational load further is currently not supported for nesting in ZoKrates, which allows scalability improvement through nesting only to a small degree of two layers.

### 8.1.1 Amount of Energy Data per Inner Proof

In the implementation, eight readings each, for daily energy consumption and production, are covered with one signature. Considering the hardware limitations of a consumer-grade computer used for the household's processing unit, such as memory and computation power, the number of signatures has to be chosen carefully. A semi-annual proof generation, as proposed in Section 6.4.1, would require 23 signatures (around 70 minutes proof generation time), which strikes a balance between computational complexity and extensive coverage.

### 8.1.2 Number of Smart Meters per Inner Proof

Theoretically, each signature can originate from a different smart meter, as each smart meter has its own signature. In doing so, one can estimate how many smart meters can be utilised for one inner proof. In this thesis use case, one or two smart meters per household seem sufficient to capture all required energy data, depending on if a second smart meter is necessary to measure energy production separately. Adding more smart meters to a household adds more costs by maintaining and managing them, than supporting the system with more data. On the other hand, in case of a larger entity, such as a factory building, it would make sense to add more smart meters to capture all energy consumption and production from different parts of the area, however, this requires more research as this thesis' focus is on households.

## 8.2 Outer Proof's Scalability

The outer proof's performance is influenced by two factors: the amount of inner proofs and the amount of public provided parameters inside each inner proof.

A linear increase in constraints per number of inner proofs can be seen in Figure 8.2. Again, the computation time increases linear until eight inner proofs and accelerates with each additional inner proof afterwards. This highlights again that ZoKrates is not scalable for an arbitrary large number of constraints. The exponential trend is clearly evident here as well, showing an even more accelerated growth in the outer proof's computation time.

Figure 8.3 demonstrates that an increase of an inner proof's public inputs leads to a linear increase in the outer proof's constraints. Here, only one inner proof is considered, which does not account for the cumulative effect across all inner proofs. Consequently,

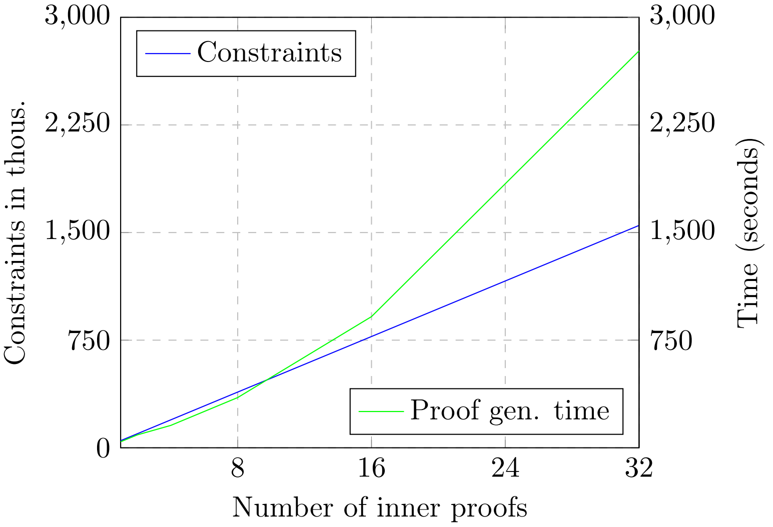

Figure 8.2: Comparison of constraints and proof generation time for various numbers of inner proofs

the increase in constraints can be represented as $n \times i$, where $n$ is the number of inner proofs and $i$ is the array size for the public inputs. Figure 8.3 shows that an inner proof with two public inputs increases the constraints of the outer proof by approximately 1.11 times, and with four public inputs, by 1.32 times, compared to an inner proof with only one public input.

Another key takeaway from this evaluation is that the inner proof can be arbitrarily complex, without influencing the outer proof's performance, as long as the proof's input array size stays the same. For example, a doubling of the inner proof's constraints has no effect on the outer proof's constraints.

### 8.2.1 Estimating the Proof Generation Time

In this section I will try to estimate the proof generation time for a higher number of inner proofs. Since the compilation and setup are only executed once, they will not be considered in this estimation. However, the witness generation time would have to be included, which is around 10 to 20 percent of the proof generation time. But for the sake of simplicity and demonstrability only the proof generation time is estimated here. The values used in this estimation are derived from those generated with the same consumer-grade laptop as mentioned previously and may improve with better hardware.

Eberhardt et al. [23] aimed for a proof generation time under 15 minutes for their netting proof with ZoKrates. However, in their system, the netting algorithm and proof generation require a more frequent execution. In contrast, the system proposed in this thesis needs to be executed less frequent. Assuming that the local utility has no hardware restrictions, unlike the smart meters or households, the proof generation time can extend multiple hours or even days.

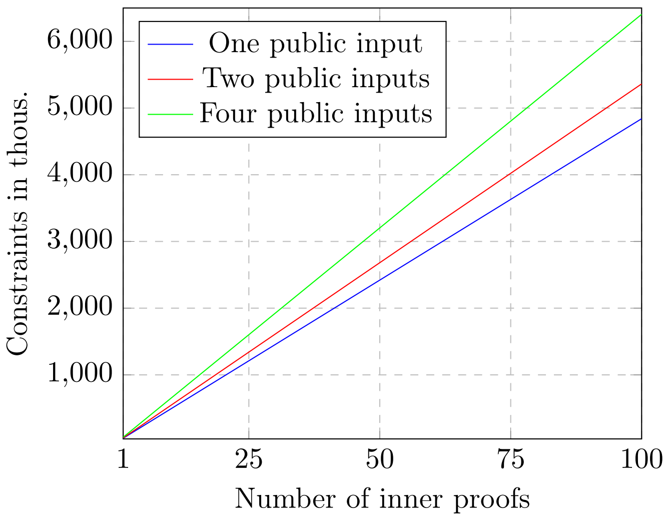

Figure 8.3: Outer proof's constraints per number of inner proofs with different amounts of public inputs

Given that the proof generation only scales linearly until eight inner proofs, an exponential growth has to be assumed. From 16 to 32 inner proofs, the base $a$ of the proof generation time is 2.6 and increases for 1.16 each doubling of inner proofs, as observed from eight to 16 and 16 to 32. Assuming that the growth factor of $a$ does not change, we can estimate the proof generation time (in seconds) for a given amount of inner proofs. I therefore modified the core formula for exponential growth , $y = a^x$, for a scenario with a doubling of $x$ and with a multiplicative increase of $y$. In doing so logarithms are applied to convert the multiplicative relationships into additive ones, which can then be modelled through exponents. As the exponential growth does not start at $x = 0$, the formula is adjusted to account for an arbitrary starting point. Here, $x$ represents the number of inner proofs, while $y$ represents the proof generation time of the outer proof and $y_0$ represent the outer proof's generation time for the starting point $x_0 = 32$, which is the maximum extent of my evaluation's computation.

$$y = y_0 \cdot a^{\log_2\left(\frac{x}{x_0}\right)}$$

$$9709 = 2766 \cdot 3,51^{\log_2\left(\frac{64}{32}\right)}$$

$$45864 = 2766 \cdot 4,07^{\log_2\left(\frac{128}{32}\right)}$$

With around 43 inner proofs, a proof generation time of approximately one hour is reached. The witness generation time accounts on top of that for about 10 to 20 percent of the proof generation time. Therefore, in realistic numbers, even fewer inner proofs can be nested for a computation time of one hour. Three hours of proof generation time allow for 96 inner proofs, assuming that the growth factor of $a$ does not accelerate. Figure 8.4 illustrates the continuation of the exponential trend for the proof generation time based on the introduced calculation.

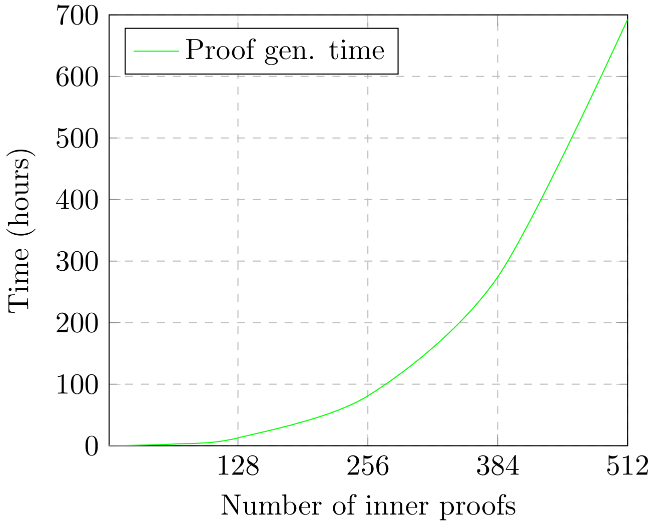

Figure 8.4: Estimating proof generation time per number of inner proofs

To better illustrate the requirements for a use case as presented in this thesis, one district or neighbourhood within an eight-digit postal code area in Germany averages around 500 households [51]. The proof generation time for 512 households is approximately 29 days, assuming no other failures, such as memory constraints. Furthermore, the compiled circuit would have a size of around 264 GB and the proving key a size of around 51 GB, assuming a linear growth in those. This highlights ZoKrates scalability limits again, as such a system implemented in ZoKrates may accommodate only smaller communities and neighbourhoods.

### 8.2.2 Number of Inner Proofs per Outer Proof

Assuming that no other factors lead to a failing of the computation, such as memory limitations, a computation time of 29 days for 512 inner proofs could potentially be feasible, considering that this has to be executed once semi-annual and that the aggregator (here local utility) has performant hardware or cloud computing capabilities. However realistically, this will probably not be feasible and future research could explore and assess the boundaries of high-performance hardware to determine the extent to which it can handle a large amount of constraints in ZoKrates.

# 9 Conclusion and Outlook

In this thesis, I have a proposed a ZoKrates-based proof of concept for enabling verifiable collective policy compliance in local renewable energy markets while preserving household privacy using nested ZKPs. Furthermore, I have illustrated how the existing body of research provides valuable insights into privacy-preserving techniques across various fields, but leaves a distinct gap in their application to policy compliance in local renewable energy markets. Additionally, the potential of nesting of ZKPs remains largely unexplored in practical applications.

Acknowledging this gap in current research, I implemented a nested ZKP system to create a verifiable and collective proof of compliance for communities. This is achieved by aggregating ZKPs from individual households in a neighbourhood, which ensure the correctness and integrity of their energy readings, into one single comprehensive proof. These inner proofs are then verified collectively within the single outer proof, which encompasses the entire adherence to the compliance standard of the community. In the end, this ZoKrates-based proof of concept has been evaluated in terms of scalability and performance to assess its limitations.

## 9.1 Conclusion

The conclusion can be succinctly divided into three aspects: on the negative side, we encounter limitations within nesting with ZoKrates, while on the positive side, the abstract concepts of privacy-preserving compliance and potential applications of nesting appear promising.

### 9.1.1 Technical Limitations of Nesting in ZoKrates

In this thesis, the exploration of ZoKrates for nested ZKPs reveals significant practical limitations in its current state, which render this approach unready for real world applications. Four major problems emerge:

First ZoKrates is limited in terms of scalability as the computation time increases exponentially with an increasing number of constraints. This not just affects the use of ZoKrates for nesting, but for general purpose. Especially in embedded systems, where some components are resource constrained, such as smart meters, it can lead to bottlenecks or the requirement of additional components to outsource the computation, such as a processing unit introduced in this thesis. Furthermore this leads to limitations in input and logic a ZoKrates ZKP can process.

The second problem is, that ZoKrates only allows for two-layered nesting (one outer proof only) in its current state. This limits the possibility of enhancing the system's scalability by adding more layers and splitting computations into smaller loads. Additionally, deeper hierarchical designs are not possible, which limits the modularity and manageability of embedded systems utilising ZoKrates.

Third, a trustless scenario is not possible as of now, as the elliptic curve, which is utilised for nesting, is not supported by the MPC protocol in ZoKrates. This leads to the reliance on a trusted setup, which minimises its potential applications.

Lastly, the lack of blockchain support for elliptic curves utilised for nesting in ZoKrates limits the potential for integrating proofs into blockchain-based embedded systems in order to improve auditability and decentralisation.

### 9.1.2 Nesting as a Promising Concept

However, dismissing those technical limitations and setting aside the concrete implementation in ZoKrates, the more abstract part of this thesis provides valuable insights. In exploring and applying the concept of nesting to compliance in local renewable energy markets, this thesis demonstrates that nesting can address use cases outside of the digital asset and blockchain sector. Nesting appears to be a promising and alternative approach for privacy-preserving data aggregation systems and protocols. With nesting, more complex privacy-preserving condition validation on collective and aggregated data is possible, simultaneously allowing for modularity and distribution in systems. Despite the technical limitations that render this thesis' implementation unsuitable for real-world scenarios, the core concepts are viable.

### 9.1.3 Privacy-Preserving Compliance is not a Paradox

Although the compliance system introduced in this thesis is not applicable to real world scenarios, as it operates under an overly optimistic best-case scenario and does not face the volatile real-world challenges, it demonstrates an essential principle: verifying collective adherence to compliance standards and regulations without compromising on privacy. Moreover, by digitising the traditional compliance process with extensive data collection and state monitoring, it reduces the reliance on physical inspections, making the process itself less intrusive.

In contrast, this approach sets a precedent for designing regulations and rules that protect individual privacy while still fulfilling necessary societal functions. It can be seen as a small yet potentially significant step towards research aimed at reducing state intrusion, while simultaneously enabling it to fulfil its societal tasks. This can support the protection of our privacy rights and democratic norms [3]. Furthermore, this approach aligns with the global trend towards more and effective compliance, as underscored by the UN report [4], and aims to reduce the privacy challenges that might be associated with this trend. By demonstrating that privacy and regulatory compliance is not necessarily paradoxical, the implementation of such systems can potentially shift the public perception towards greater acceptance and cooperation in regulation. Ultimately, this supports the achievement of societal objectives, as regulatory measures are viewed not as intrusive, but as respectful of individual rights. This can enhance the sustainability of our society, not just in preserving our rights but also in improving the effectiveness of environmental compliance. In doing so, climate change can be targeted more effectively.

## 9.2 Outlook and Future Work

Future work might enhance this thesis by applying the nested proof structure for collective compliance verification across different ZKP frameworks and alternative types of ZKPs, such as STARKs or PLONKs. For example, Mina's Kimchi [52] or Polygon's PIL toolkit [53] seem to be promising in this regard, as they even allow for an arbitrary number of layers in nesting and recursion, potentially enhancing the scalability compared to ZoKrates by far. Having multiple different implementations of this thesis' concept with various state of the art ZKP toolkits enables more comprehensive benchmarking and performance comparisons. Moreover, future research might enhance the in here proposed system to face the volatile real-world challenges, making the system real-world applicable. By utilising more efficient ZKP types, it may be feasible to run them directly on smart meter hardware, improving the use of ZKPs in embedded system significantly. In doing so, the dependency on households having a processing unit can be eliminated, making the system more robust.

Regarding ZoKrates, future work might implement the elliptic curves required for nesting into its MPC in order to decentralise the setup phase. Alternatively, additional elliptic curves might be implemented, which improve the performance of ZoKrates and enable further layers for nesting.

# List of Acronyms

| | |
|---|---|
| CRS | Common Reference String |
| HPU | Household Processing Unit |
| MPC | Multi-Party Computation Protocol |
| PDA | Privacy-Preserving Data Aggregation |
| PLONK | Permutations over Lagrange-bases for Oecumenical Noninteractive arguments of Knowledge |
| WDS | Water Distribution System |
| zk-SNARKS | zero-knowledge Succinct Non-interactive Arguments of Knowledge |
| zk-STARK | zero-knowledge Scalable Transparent Argument of Knowledge |
| ZKP | Zero Knowledge Proof |

# Bibliography

[1] "Energieverbrauch privater haushalte," Umweltbundesamt, 2023, accessed: 28-12-2023. [Online]. Available: https://www.umweltbundesamt.de/daten/private-haushalte-konsum/wohnen/energieverbrauch-privater-haushalte

[2] A. Molina-Markham, P. Shenoy, K. Fu, E. Cecchet, and D. Irwin, "Private memoirs of a smart meter," in *Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*, ser. BuildSys '10. New York, NY, USA: Association for Computing Machinery, 2010, p. 61–66. [Online]. Available: https://doi.org/10.1145/1878431.1878446

[3] K. Eck and S. Hatz, "State surveillance and the covid-19 crisis," *Journal of Human Rights*, vol. 19, no. 5, pp. 603–612, 2020. [Online]. Available: https://doi.org/10.1080/14754835.2020.1816163

[4] United Nations Environment Programme (UNEP), "Environmental rule of law: First global report," United Nations Environment Programme, Nairobi, Tech. Rep., 2019, iSBN: 978-92-807-3742-4. [Online]. Available: https://www.unep.org/resources/publication/environmental-rule-law-tracking-progress-and-charting-future-directions

[5] "Etymology of "privacy"," Online Etymology Dictionary, 2024, accessed: 02-01-2024. [Online]. Available: https://www.etymonline.com/word/privacy

[6] A. Alibeigi, A. B. Munir, and M. E. Karim, "Right to privacy, a complicated concept to review," *SSRN*, 2019. [Online]. Available: https://doi.org/10.2139/ssrn.3537968

[7] L. S. Strickland and L. E. Hunt, "Technology, security, and individual privacy: New tools, new threats, and new public perceptions," *Journal of the American Society for Information Science and Technology*, vol. 56, no. 3, pp. 221–234, 2005. [Online]. Available: https://doi.org/10.1002/asi.20122

[8] M. A. Mustafa, S. Cleemput, and A. Abidin, "A local electricity trading market: Security analysis," in *2016 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, 2016, pp. 1–6. [Online]. Available: https://doi.org/10.1109/ISGTEurope.2016.7856269

[9] A. Acquisti, L. Brandimarte, and G. Loewenstein, "Privacy and human behavior in the age of information," *Science*, vol. 347, no. 6221, pp. 509–514, 2015. [Online]. Available: https://doi.org/10.1126/science.aaa1465

[10] C. Cuijpers and B.-J. Koops, "Het wetsvoorstel 'slimme meters': een privacytoets op basis van art. 8 evrm," *Universiteit van Tilburg*, 2008. [Online]. Available: https://www.researchgate.net/publication/254796082_Het_wetsvoorstel_%27slimme_meters%27_Een_privacytoets_op_basis_van_art_8_EVRM

[11] F. D. Garcia and B. Jacobs, "Privacy-friendly energy-metering via homomorphic encryption," in *Security and Trust Management*, J. Cuellar, J. Lopez, G. Barthe, and A. Pretschner, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 226–238. [Online]. Available: https://doi.org/10.1007/978-3-642-22444-7_15

[12] "Compliance," Cambridge Dictionary, 2024, accessed: 02-01-2024. [Online]. Available: https://dictionary.cambridge.org/dictionary/english/compliance

[13] A. Karakostas and D. J. Zizzo, "Compliance and the power of authority," *Journal of Economic Behavior  Organization*, vol. 124, pp. 67–80, 2016, taxation, Social Norms and Compliance. [Online]. Available: https://doi.org/10.1016/j.jebo.2015.09.016

[14] H. Kyngäs, M. Duffy, and T. Kroll, "Conceptual analysis of compliance," *Journal of clinical nursing*, vol. 9, no. 1, p. 5—12, January 2000. [Online]. Available: https://doi.org/10.1046/j.1365-2702.2000.00309.x

[15] O. Goldreich, S. Micali, and A. Wigderson, "Proofs that yield nothing but their validity or all languages in np have zero-knowledge proof systems," *J. ACM*, vol. 38, no. 3, p. 690–728, jul 1991. [Online]. Available: https://doi.org/10.1145/116825.116852

[16] M. Petkus, "Why and how zk-snark works," *CoRR*, vol. abs/1906.07221, 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1906.07221

[17] E. Ben-Sasson, A. Chiesa, E. Tromer, and M. Virza, "Scalable zero knowledge via cycles of elliptic curves," *Algorithmica*, vol. 79, pp. 1102–1160, 2017. [Online]. Available: https://doi.org/10.1007/s00453-016-0221-0

[18] S. Bowe, J. Grigg, and D. Hopwood, "Recursive proof composition without a trusted setup," Cryptology ePrint Archive, Paper 2019/1021, 2019, https://eprint.iacr.org/2019/1021. [Online]. Available: https://eprint.iacr.org/2019/1021

[19] P. Valiant, "Incrementally verifiable computation or proofs of knowledge imply time/space efficiency," in *Theory of Cryptography*, R. Canetti, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 1–18. [Online]. Available: https://doi.org/10.1007/978-3-540-78524-8_1

[20] O. Astrachan, "Self-reference is an illustrative essential," in *Proceedings of the Twenty-Fifth SIGCSE Symposium on Computer Science Education*, ser. SIGCSE '94. New York, NY, USA: Association for Computing Machinery, 1994, p. 238–242. [Online]. Available: https://doi.org/10.1145/191029.191131

[21] J. Eberhardt and S. Tai, "Zokrates - scalable privacy-preserving off-chain computations," in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2018, pp. 1084–1091. [Online]. Available: https://doi.org/10.1109/Cybermatics_2018.2018.00199

[22] J. Heiss, T. Oegel, M. Shakeri, and S. Tai, "Verifiable carbon accounting in supply chains," *IEEE Transactions on Services Computing*, pp. 1–14, 2023. [Online]. Available: https://doi.org/10.1109/TSC.2023.3332831

[23] J. Eberhardt, M. Peise, D.-H. Kim, and S. Tai, "Privacy-preserving netting in local energy grids," in *2020 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, 2020, pp. 1–9. [Online]. Available: https://doi.org/10.1109/ICBC48266.2020.9169440

[24] G. Ismayilov and C. Ozturan, "Trustless privacy-preserving data aggregation on ethereum with hypercube network topology," *arXiv preprint arXiv:2308.15267*, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2308.15267

[25] Y. El Housni, M. Connor, A. Guillevic, and hujw77, "Eip-3026: Bw6-761 curve operations," 2020, accessed: 25-01-2024. [Online]. Available: https://eips.ethereum.org/EIPS/eip-3026

[26] F. W. Dekker and Z. Erkin, "Privacy-preserving data aggregation with probabilistic range validation," in *Applied Cryptography and Network Security*, K. Sako and N. O. Tippenhauer, Eds. Cham: Springer International Publishing, 2021, pp. 79–98. [Online]. Available: https://doi.org/10.1007/978-3-030-78375-4_4

[27] A. Mohammadali and M. S. Haghighi, "A privacy-preserving homomorphic scheme with multiple dimensions and fault tolerance for metering data aggregation in smart grid," *IEEE Transactions on Smart Grid*, vol. 12, no. 6, pp. 5212–5220, 2021. [Online]. Available: https://doi.org/10.1109/TSG.2021.3049222

[28] L. Zhang, J. Zhang, and Y. H. Hu, "A privacy-preserving distributed smart metering temporal and spatial aggregation scheme," *IEEE Access*, vol. 7, pp. 28 372–28 382, 2019. [Online]. Available: https://doi.org/10.1109/ACCESS.2019.2899961

[29] A. Rial and G. Danezis, "Privacy-preserving smart metering," in *Proceedings of the 10th Annual ACM Workshop on Privacy in the Electronic Society*, ser. WPES '11. Association for Computing Machinery, 2011, p. 49–60. [Online]. Available: https://doi.org/10.1145/2046556.2046564

[30] M. Jawurek, M. Johns, and F. Kerschbaum, "Plug-in privacy for smart metering billing," in *International Symposium on Privacy Enhancing Technologies Symposium.* Springer, 2011, pp. 192–210. [Online]. Available: https://doi.org/10.1007/978-3-642-22263-4_11

[31] D. Mouris and N. G. Tsoutsos, "Masquerade: Verifiable multi-party aggregation with secure multiplicative commitments," Cryptology ePrint Archive, Paper 2021/1370, 2021. [Online]. Available: https://eprint.iacr.org/2021/1370

[32] H. H. M. Mahmoud, W. Wu, and Y. Wang, "Secure data aggregation mechanism for water distribution system using blockchain," in *2019 25th International Conference on Automation and Computing (ICAC)*. IEEE, 2019, pp. 1–6. [Online]. Available: https://doi.org/10.23919/IConAC.2019.8895146

[33] Z. Guan, G. Si, X. Zhang, L. Wu, N. Guizani, X. Du, and Y. Ma, "Privacy-preserving and efficient aggregation based on blockchain for power grid communications in smart communities," *IEEE Communications Magazine*, vol. 56, no. 7, pp. 82–88, 2018. [Online]. Available: https://doi.org/10.1109/MCOM.2018.1700401

[34] H. Fan, Y. Liu, and Z. Zeng, "Decentralized privacy-preserving data aggregation scheme for smart grid based on blockchain," *Sensors*, vol. 20, no. 18, p. 5282, 2020. [Online]. Available: https://doi.org/10.3390/s20185282

[35] T. Miyamae, F. Kozakura, M. Nakamura, S. Zhang, S. Hua, B. Pi, and M. Morinaga, "Zgridbc: Zero-knowledge proof based scalable and private blockchain platform for smart grid," in *2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, 2021, pp. 1–3. [Online]. Available: https://doi.org/10.1109/ICBC51069.2021.9461122

[36] M. Kirschbaum, T. Plos, and J.-M. Schmidt, "On secure multi-party computation in bandwidth-limited smart-meter systems," in *2013 International Conference on Availability, Reliability and Security*, 2013, pp. 230–235. [Online]. Available: https://doi.org/10.1109/ARES.2013.137

[37] C. Thoma, T. Cui, and F. Franchetti, "Secure multiparty computation based privacy preserving smart metering system," in *2012 North American Power Symposium (NAPS)*, 2012, pp. 1–6. [Online]. Available: https://doi.org/10.1109/NAPS.2012.6336415

[38] G. Eibl and D. Engel, "Differential privacy for real smart metering data," *Computer Science-Research and Development*, vol. 32, pp. 173–182, 2017. [Online]. Available: https://doi.org/10.1007/s00450-016-0310-y

[39] P. Barbosa, A. Brito, H. Almeida, and S. Clauß, "Lightweight privacy for smart metering data by adding noise," in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, ser. SAC '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 531–538. [Online]. Available: https://doi.org/10.1145/2554850.2554982

[40] E. Morais, T. Koens, C. Van Wijk, and A. Koren, "A survey on zero knowledge range proofs and applications," *SN Applied Sciences*, vol. 1, pp. 1–17, 2019. [Online]. Available: https://doi.org/10.1007/s42452-019-0989-z

[41] "The go-fast machine: Adding recursion to polygon zkevm," Polygon Labs, accessed: 13-01-2024. [Online]. Available: https://polygon.technology/blog/the-go-fast-machine-adding-recursion-to-polygon-zkevm

[42] "Lightweight blockchain - mina protocol," Mina Foundation, accessed: 13-01-2024. [Online]. Available: https://minaprotocol.com/lightweight-blockchain

[43] N. Verkade and J. Höffken, "Collective energy practices: A practice-based approach to civic energy communities and the energy system," *Sustainability*, vol. 11, no. 11, 2019. [Online]. Available: https://doi.org/10.3390/su11113230

[44] F. Moret and P. Pinson, "Energy collectives: A community and fairness based approach to future electricity markets," *IEEE Transactions on Power Systems*, vol. 34, no. 5, pp. 3994–4004, 2019. [Online]. Available: https://doi.org/10.1109/TPWRS.2018.2808961

[45] R. T. Rust, P. K. Kannan, and N. Peng, "The customer economics of internet privacy," *Journal of the Academy of Marketing Science*, vol. 30, pp. 455–464, 2002. [Online]. Available: https://doi.org/10.1177/009207002236917

[46] A. Acquisti, C. Taylor, and L. Wagman, "The economics of privacy," *Journal of Economic Literature*, vol. 54, no. 2, pp. 442–92, June 2016. [Online]. Available: https://doi.org/10.1257/jel.54.2.442

[47] Z. Mani and I. Chouk, "Impact of privacy concerns on resistance to smart services: does the 'big brother effect' matter?" *Journal of Marketing Management*, vol. 35, no. 15-16, pp. 1460–1479, 2019. [Online]. Available: https://doi.org/10.1080/0267257X.2019.1667856

[48] S.-C. Yip, W.-N. Tan, C. Tan, M.-T. Gan, and K. Wong, "An anomaly detection framework for identifying energy theft and defective meters in smart grids," *International Journal of Electrical Power Energy Systems*, vol. 101, pp. 189–203, 2018. [Online]. Available: https://doi.org/10.1016/j.ijepes.2018.03.025

[49] J. Eberhardt, "Mock sensor for decentralized energy trading," 2019, accessed: 13-02-2024. [Online]. Available: https://github.com/JacobEberhardt/decentralized-energy-trading/tree/master/mock-sensor

[50] R. Mehrpoya, "Bachelor thesis," 2023, accessed: 04-02-2024. [Online]. Available: https://github.com/uZhW8Rgl/Bachelor-Thesis

[51] "PLZ8 Deutschland Grenzen PLZ8 Deutschland XXL Release," PTV Group, 2022, accessed: 03-03-2024. [Online]. Available: https://www.ptvgroup.com/sites/default/files/2022-09/PTV_PLZ8_2210_DB.pdf

[52] "Kimchi - Mina book," Mina Foundation, 2023, accessed: 02-03-2024. [Online]. Available: https://o1-labs.github.io/proof-systems/specs/kimchi.html

[53] “pil-stark: Generates a stark from a pil,” Polygon Labs, 2023, accessed: 02-03-2024. [Online]. Available: https://github.com/0xPolygonHermez/pil-stark

# Annex

Listing 1: ZoKrates code of the inner proof of integrity with two signatures

```

from "ecc/decaf377.zok" import verifyEddsa;
import "utils/casts/u32_to_field.zok" as u32_to_field;

def main(private field[2] R1, private field S1, private field[2] R2,
    private field S2, private field[2] A, private u32[16] Consumption,
    private u32[16] Production, field netResult){
    assert(verifyEddsa(R1, S1, A, Consumption[0..8], Production[0..8]));
    assert(verifyEddsa(R2, S2, A, Consumption[8..16], Production[8..16]));

    u32 mut sumConsumption = 0;
    u32 mut sumProduction = 0;

    for u32 i in 0..16 {
        sumConsumption = sumConsumption + Consumption[i];
        sumProduction = sumProduction + Production[i];
    }
    field fieldSumConsumption = u32_to_field(sumConsumption);
    field fieldSumProduction = u32_to_field(sumProduction);
    field calculatedNetResult = fieldSumConsumption - fieldSumProduction;
    assert(calculatedNetResult == netResult);

}
```

Listing 2: ZoKrates code of the outer proof of compliance

```

from "snark/gm17" import main as verify, Proof, VerificationKey;

const u32 PROOF_INPUTS = 1;
const u32 VERIFICATION_KEY_SIZE = PROOF_INPUTS + 1;
const u32 NUM_PROOFS = 4;
const field prime_bs =
    8444461749428370424248824938781546531375899335154063827935233455917409239041;


struct PrivateInputs {
    Proof<PROOF_INPUTS>[NUM_PROOFS] proofs;
    VerificationKey<VERIFICATION_KEY_SIZE>[NUM_PROOFS] keys;
}

def convert_negative_num_from_bls_to_bw(field bls_num) -> field{
    field diff = prime_bs - bls_num;
    field bw_num = 0 - diff;
    return bw_num;
}

def main(private PrivateInputs privateInputs, field limit) {
    field mut totalNetResult = 0;

    for u32 i in 0..NUM_PROOFS {
        assert(verify(privateInputs.proofs[i], privateInputs.keys[i]));
        field householdNetResult = privateInputs.proofs[i].inputs[PROOF_INPUTS-1];
        totalNetResult = totalNetResult + (householdNetResult >= 100000000 ? convert_negative_num_from_bls_to_bw(householdNetResult) : householdNetResult); //assuming no householdNetResult is above 100000000
    }
    assert(totalNetResult <= limit);
}
```

Listing 3: ZoKrates function to convert a negative number from Bls3_77 to bw6_761

```
def negative_num_from_bls_to_bw(field negBlsNum) -> field{
    field primeBs = 8444...;
    field dif = primeBs - negBlsNum;
    field negBwNum = 0 - diff;
    return negBwNum;
}
```

| Signatures | Constraints | Compilation | Witness gen. | Proof gen. |
|---|---|---|---|---|
| 1 | 94237 | 20.19 s | 4.12 s | 14.94 s |
| 2 | 187556 | 41.17 s | 9.46 s | 26.19 s |
| 4 | 374192 | 85.42 s | 14.86 s | 44.08 s |
| 8 | 747462 | 201.32 s | 28.34 s | 108.04 s |
| 16 | 1493998 | 1303.18 s | 56.12 s | 285.91 s |
| 32 | 2987070 | 8338.54 s | 98.45 s | 743.29 s |

Table .1: Inner proof's constraints and computation time per amount of signatures

| Inner proof's input array size | Outer proofs's constraints with one inner proof |
|---|---|
| 0 | 42037 |
| 1 | 47254 |
| 2 | 52471 |
| 4 | 62905 |

Table .2: Outer proof's constraints per increase in the inner proof's input-array size (public inputs)

| Inner proofs | Constraints | Compilation | Witness gen. | Proof gen. |
|---|---|---|---|---|
| 1 | 49522 | 45.23 s | 7.2 s | 42.10 s |
| 2 | 97912 | 82.88 s | 21.06 s | 90.35 s |
| 4 | 194692 | 137.02 s | 30.34 s | 156.01 s |
| 8 | 388252 | 265.75 s | 47.00 s | 349.81 s |
| 16 | 775372 | 902.44 s | 98.35 s | 914.10 s |
| 32 | 1549612 | 2033.83 s | 183.21 s | 2766.07 s |

Table .3: Outer proof's constraint and computation time per inner proofs